GENERATING SEMANTICALLY SIMILAR PERMUTATIONS OF QUESTIONS

LEUNG CHEUK YUI

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2019 / 2020

NANYANG TECHNOLOGICAL UNIVERSITY

SCSE19-0007

CZ4079

GENERATING SEMANTICALLY SIMILAR PERMUTATIONS OF QUESTIONS

Submitted in Partial Fulfillment of the Requirements
for the Degree of Bachelor of Computer Science
of the Nanyang Technological University

by

LEUNG CHEUK YUI

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
2019 / 2020

Abstract

This project will introduce an effective question generation engine named, QG system. The QG system will generate semantically similar permutations of English questions with Chinese language influence which aims to improve the performance of automated question answering systems such as chatbots. This is achieved by studying the linguistic differences between English and Chinese sentences such as differences in their lexical and syntax usage. Two approaches were proposed with the first approach utilising machine translation applications and the second approach utilising the permutations of Chinese sentences. Evaluation results shows that the second approach is more effective in generating more permutations of questions using the input English question and introducing a stronger Chinese influence into the generated questions compared to the first approach.

Acknowledgements

This project is only made possible because of the help from the following people and I would like to express my utmost gratitude to them:

First and foremost, I would like to express my sincere gratitude to my supervisor, Assoc Prof. Chng Eng Siong for his guidance and providing valuable insightful feedback on the project. Amidst his tight schedule, Prof. Chng organised weekly meetings with his Final Year Project (FYP) students to update him on our progress and offering pointers to help us stay on track for the project. Under Prof. Chng's guidance, I was able to gain valuable skills in presentation of my research findings in academic settings.

Secondly, I would like to thank Mr. Andrew Koh Jin Jie and Mr. Yap Boon Peng for providing valuable feedback and resources that are beneficial to the project.

Thirdly, I would like to express my gratitude to my family and friends who have provided me support in any aspect throughout the project.

Last but not least, I would like to thank Nanyang Technological University, School of Computer Science and Engineering for providing me with this opportunity to research on this project.

List of Figures and Tables

Table of Contents

Chapter 1 Introduction

1.1 Background and Motivation

Question answering (QA) is one of the main problems faced in the field of Natural Language Processing (NLP). QA refers to the process of retrieving or generation of answers for questions imposed by human users in their native language. This is most commonly seen in the use of chatbots. Chatbots are a form of conversational Artificial Intelligence (AI) which are designed to simplify human interaction with computers. Using chatbots, computers can understand and respond to human input through spoken or written language [1].

Chatbots can be programmed to respond to simple keywords or prompts, or to hold complex conversations about specific topics. They range in complexity from information retrieval using keyword matches to active learning capabilities that provide in-depth responses and tailored suggestions based on previous conversations. Many industries use chatbots to improve or increase the efficiency of customer service and e-commerce.

Chatbots communicate through speech or text. Both rely on artificial intelligence technologies like machine learning and natural language processing. Natural language processing is a branch of artificial intelligence that teaches machines to read, analyze and interpret human language. This technology gives chatbots a baseline for understanding language structure and meaning. NLP, in essence, allows the computer to understand what you are asking and how to appropriately respond by using neural networks to train them with training data so that the best and correct answer for a given question will be selected [2].

Domain specific QA systems, such as the frequently-ask-questions (FAQ) section in government and customer service websites, often do not have enough training dataset to train a robust and accurate deep learning model. These training dataset from these FAQ systems usually consist of fixed, one-to-one mappings of question to answer. Although we can continuously obtain more training examples by generating a variety of questions through the manual thinking process, this method is expensive and time-consuming. Therefore, we proposed to have an automatic question generator, named Question Generation (QG) system, that takes a question as input and generates questions with different lexical and syntactic

structures while preserving the original meaning of the input question. These generated questions can then be used to augment the original training examples for deep learning [3].

Question generation from a given paragraph of text is widely studied by researchers and students and is one of the solutions to generate questions for the deep learning process for chatbot training. In one of the Final Year Projects done by students in previous years from Nanyang Technological University (NTU), Kurniawan Aryanto Famili, clustering was the method used to generate different permutations of questions to train the chatbot for the NTU FAQ website [4].

However, Kurniawan concluded his research with a limitation which is that there is a lack in quality of the question permutations. Therefore, my proposed project will include other more robust methods to generate a variety of semantically similar questions based on one input question. In addition to generating different questions, we will also evaluate the accuracy of the output questions compared to the input question using various metrics to determine the fluency, soundness of the sentence and whether the meaning is preserved for the output questions.

English language may be the most spoken language in Singapore as English language and the mother tongue language of all children are taught since they are in preschool. As of end-June 2019, Chinese residents composed a percentage of 74.4% of Singapore's population [5]. This gives rise to the number of Singaporeans who are bilingual in both English and Chinese language. A government survey done in 2016 shows that English language is the most common language spoken at home with a percentage of 36.9% with Chinese language close behind with a percentage of 34.9% [6]. This shows a large percentage of the population that continue to use Chinese language as a means of their daily communication. While these Chinese speakers are able to communicate in English language effectively and formally when needed, their English sentences for daily communication would still have the presence of the Chinese language influence and other common dialects due to the simplicity and ease of understanding among Singaporeans. Therefore, there is a need to explore ways to generate permutations of English questions with Chinese influence for the training of the question answering model for the FAQ chatbot meant for Singaporeans.

1.2 Objectives

The primary goal of this project is to implement a question generation engine capable of generating a wide variety of questions that can be used to augment the training datasets of text-based QA tasks.

The following are the objectives of this project:

1. Implement a question generation engine
2. Adapt existing text generation and text augmentation methods to question generation
3. Evaluate performance of each question generation methods in question answering tasks
4. Perform qualitative analysis on generated questions

1.3 Scope of Work

The scope of this project is limited to data concerning commonly asked questions regarding the Baby Bonus Scheme from the Ministry of Social and Family Development (MSF). However, the developed system should be extensible and applicable to other provided data as well in the future.

1.4 Organisation of Report

This report is organised into five chapters as follow:

- Chapter 1 introduces the project and defines its objectives and scope of the project.
- Chapter 2 provides literature review on the linguistic differences between English and Chinese sentences as well as on topics and techniques related to question and text generation.
- Chapter 3 introduces the methodology that will be used in the implementation of the Question Generation (QG) system in this project.
- Chapter 4 provides the implementation details of the QG system as well as the results and analysis of the effectiveness of the QG system by analysing the generated questions.
- Chapter 5 discusses the future works and provides a conclusion to the report.

Chapter 2 Literature Review

Question generation often through question expansion generates more questions by various approaches such as re-weighting terms, stemming, replacing words with synonyms, etc. Question expansion is often required in Question Answering (QA) tasks to improve the information retrieval performance so that more questions can be trained to match the correct answer in more documents. These methods are limited to permuting the English questions. Considering other languages, e.g. Chinese language, we learn that English and Chinese sentences have different sentence structures. Inspired by this, we have the idea of generating English questions with Chinese sentence structure influence so that we can generate English questions of different permutations yet preserving the semantic meaning of the original English question. In this chapter, we discuss the differences between English and Chinese sentences that we have to consider when generating different permutations of semantically similar English questions using the structure of Chinese sentences as well as the different approaches of question generation.

2.1 Linguistic differences between English and Chinese

English language and Chinese language have many differences. While native Chinese speakers are capable of mastering English language and vice versa, Chinese influence on English sentences written or spoken by native Chinese speakers has been found to be largely present in the syntax and lexical usage of the English sentences [7]. Despite the different structure of English sentences due to Chinese language influence, native Chinese speakers are still able to understand one another.

2.1.1 Syntax

In linguistics, syntax refers to a set of rules and principles that rule the sentence structure and word order in a given language. It defines the format in which words and phrases are arranged to create coherent sentences.

The English Language is a subject-prominent language which means that almost all English sentences must have a subject and tend to center around the verb. Subject may only be represented by pronoun, noun and

noun phrase from the English language. Simple English sentences can be represented in the sentence forming formula below:

S (Sentence) = **NP** (Noun Phrase) + **VP** (Verb Phrase)

This formula can be seen in an example sentence, *"The man is doing work"*, where the noun phrase is represented by *"The man"* and the verb phrase is represented by *"doing work"*. The noun phrase, *"The man"*, represents the subject of the sentence.

The Chinese language, unlike English language, is a topic-prominent language where the need of having a subject being secondary, it is more important to have a topic and a comment related to the topic. There are no rules specifying the position of the subject in a Chinese sentence. Modern Chinese grammarians have thus suggested that a typical Chinese sentence may be represented in the following formula:

S (Sentence) = **T** (Theme) + **R** (Rheme)

Theme and *Rheme* in the formula refer to the *topic* and the *comment* related to the topic of the sentence respectively. Topics can be classified into six categories, namely the NP (noun phrase), S (sentence), S' (topic sentence), VP (verb phrase), Prep. P (prepositional phrase) and Post. P (post-positioned phrase). They are listed in the following example sentences:

1. 这些话我不会相信。(NP)
2. 他会说这些话我不会相信。(S)
3. 这些话他会说我不会相信。(S')
4. 在桌上他放了几本书。(Prep. P)
5. 桌上有几本书，床上不会有书。(Post. P)
6. 说这些话我赞成。(VP)

The underlined parts in the above sentences are *topics* of the sentences and categorised into 6 classes as shown above.

2.1.2 Lexical Usage

Belonging to two different language families, English and Chinese has significant differences in their lexical usage. The Chinese language does not have an alphabetic system but instead uses a logographic system for its written language. In logographic systems, the symbols represent the words themselves as compared to English words which are made up of various letters from the alphabetic system.

The use of English words has grown through the expansion of the base meaning of the base word. A base word can be given new meaning through methods such as extension or using a metaphor. The following figure shows an example that exhibit this behaviour:

Word	Meaning	Part of speech	Remarks
Sign	A mark	Noun	Base word
Signify	To make a mark	Verb	Affix added to the word
Significant	Making a mark	Adjective	Affix added to the word
Insignificant	Making no mark	Adjective	Prefix added to the word
Insignificance	The making of no mark	Noun	Suffix added to the word

Figure 2-1: Morphology of the word Sign

For the Chinese lexicon, unlike English lexicons, has no change made to the word formations. The formation of new phrases of new meanings is through the combination of two or more Chinese characters to form a new single word phrase. This can be seen the examples below:

- 水 + 壶 = 水壶 (Water + Bottle = Water Bottle)
- 电 + 脑 = 电脑 (Electric + Brain = Computer)

The difference in English and Chinese lexicons results in differences when translating the sentences from one language to the other.

2.2 Question Answering

In the field of question generation for question answering, researchers from a recent study [8] categorised question answering systems into four groups:

1. **Community-based QA**: This is where questions and answers are posted and contributed by online communities. Users who have a question they want to ask will post the question on an online question answering platform such as Yahoo! Answers and other users will contribute answers to the question directly.
2. **Knowledge-based QA**: This is where answers are stored in a structured form (e.g. President (USA, Donald Trump)) in a large database such as Freebase. Before the user is able to query answers from the knowledge base, questions written by users in natural language will first have to be converted to a structured query that is understandable to the knowledge base.
3. **Text-based QA**: This is where a list of ready-made answers is used to match to a question queried in natural language form. The answer which returns the highest matching score will be chosen as the best answer to the question.
4. **Reading comprehension**: This is where answers are generated from a paragraph of texts based on the question asked. This system is often regarded as the most challenging task in question answering as the reading comprehension system must be very robust such that it is able to fully comprehend the question and the paragraph of texts so that the correct answer can be extracted from the paragraph of texts.

In this thesis, text-based QA will be focused as the question generation system proposed in this thesis is meant for FAQ answering. Recently, there are increasingly more organisations converting their static FAQ answering sections into chatbots allowing users to have more interaction when having a query. Having chatbots allows users to ask a question in natural language form and the system will select and return the best answer to the user. This way users would not have to spend time to look for the answer to their queries by reading through the entire static FAQ section. Chatbots can be treated as automatic FAQ answering systems where it involves the task of selecting the best answers in a list of static candidate answers based on the query. Therefore, there is a need to train a robust FAQ answering system with a large database of a variety of questions so that the system is able to recognise and understand the question so as to choose the best and matching answer to the question. Hence, this thesis will investigate new approaches to generate a variety of questions.

2.3 Question Generation

Traditionally, Question Generation (QG) refers to the process of generating questions related to given inputs such as answers to a question or a paragraph of text [9]. QG has many real world applications, such as question generation for exams and assessments where questions are generated from the text content, dialogue generations in chatbots and datasets expansion in Question Answering tasks. The input for QG can be in different forms. Traditionally, QG takes in input in textual forms in the form of sentences or a paragraph of text. There are also other researches in QG that take in non-textual inputs for the process of QG such as images [10] and knowledge base [11]. In this thesis, we continue to focus on taking in textual inputs for our question generation. In contrast to the traditional methods that use answers or paragraphs of texts to generate questions, we aim to use only a single question to generate semantically similar permutations of that question. The underlying motivation is that the traditional methods of QG aim to create different types of questions based on the given input text, but our goal is to explore ways to generate multiple permutations of questions similar to the input question. The traditional methods include using rule-based approach and implementation of neural networks to generate questions.

2.3.1 Rule-based Approach

Early research done in QG focuses more on rule-based QG. A recent survey paper about QG categorised rule-based QG [12] into two different approaches, namely the transformation-based [13] and template-based [14] approach. The transformation-based approach transforms the given input at surface level to generate questions while the template-based approach uses pre-defined templates to generate questions.

Generation of questions with the rule-based approach involves defining context-free grammar which is a set of rules that needs to be followed to generate all possible types of sentences or strings that fulfill the rules that are set. In order to consider all possible combinations of sentences, a large and comprehensive set of rules must be handcrafted in this rule-based approach.

In previous researches, a large set of rules are handcrafted to identify the key purpose of the question so that the semantic meaning of the question can be preserved while replacing the question words to form questions in different formats that continue to retain the semantic meaning of the original input question. However, handcrafting the set of context-free grammar for the generation of questions can be tedious for more complicated and longer questions and hence, might not be able to cover all possible patterns of

questions. In conclusion, in the long run, rule-based approach might work well for simple and short questions but for more complicated questions, it might be an ineffective method to generate different permutations of that question.

2.3.2 Neural Networks

Since the advancement of deep learning in Natural Language Processing (NLP), neural networks are gaining popularity to be used in generating questions. Most Neural Question Generation (NQG) models proposed [15] are based on the Seq2Seq architecture [16], which encodes the given textual input such as sentences or paragraphs of text into vector representations and then generating the questions by decoding the encoded vectors. In previous research, two different approaches using neural networks were proposed, namely a retrieval-based approach using convolutional neural networks (CNN) and a generation-based approach using recurrent neural networks (RNN).

In order to begin with neural networks, a dataset of all the related questions must be gathered to mine the different question patterns and identify the topics that are related to the questions. From there, neural networks will be used to predict the question pattern and identify the topic that the question is about and then generate different permutations of the question patterns. This prediction can be done by either using the retrieval-based approach or the generation-based approach. The retrieval-based approach ranks the frequency of the different types of question patterns and selects the question patterns that are highly ranked. On the other hand, the generation-based approach generates question patterns using the Seq2Seq architecture which can generate new question patterns from the original question patterns mined from the original dataset. After producing the new question patterns and identifying the topic of the question, the topic of the question will then be fitted into the new question pattern to generate new permutations of questions.

Attention-based sequence learning model was also proposed as another method using neural networks to generate questions [17]. This model is implemented using the encoder-decoder neural networks architecture with an attention mechanism which uses a bidirectional Long Short Term Memory (LSTM) for the encoding layer to encode the given paragraph of texts into vector representations. The bidirectional LSTM improves the subtask of selecting the key sentence and together with the attention-based model, the model is able to generate questions related to the given textual input. However, the performance of this model is limited as it requires a large training dataset in order to have better performance.

2.4 Neural Machine Translation

Neural machine translation (NMT) has become widely used in many machine translation systems ever since it was introduced in 2013 [18]. Similar to neural networks, NMT models are also based on the Seq2Seq architecture involving an encoder and a decoder. In order to translate a sentence, the encoder will first have to encode the textual input, for example a sentence in English language, into single vector representation and the decoder will then have to decode the encoded vector representation to generate a sentence in the designated language to be translated to.

One method of the NMT approach of generating questions is to train a NMT model using a corpus of paraphrased sentences from the source sentence of a single language. This NMT model will then use the Seq2Seq architecture of encoder and decoder to encode the input question into vector representations. After which, the decoder will decode the encoded sentence as if the model is translating the input question into another language when in fact it is changing the structure of the input question to form new permutations of questions while preserving the semantic meaning. However, this approach requires a large corpus to be able to train the model well enough to have significantly good results.

2.5 Text Augmentation

Text augmentation refers to the task of tokenization of sentences or paragraphs of texts into smaller components for augmentation back into sentences and often shuffling these smaller components and rejoining them to form new sentences or paragraphs. This approach can be used in question generation to augment input questions to form new questions. A simple yet effective data augmentation technique for text data was proposed by Wei, Jason W., and Kai Zou [19]. This technique is called Easy Data Augmentation (EDA) and it consists of 4 components namely, synonym replacement, random insertion, random swapping and random deletion.

The full details of each component of the EDA is introduced below:

1. **Synonym Replacement (SR)**: Choose *n* words from the sentence randomly that are not stop words. Replace each of these words with one of its synonyms chosen at random.

2. **Random Insertion (RI)**: For a random word in the sentence that is not a stop word, find a corresponding synonym for that word. Insert that synonym into a random position in the sentence. Repeat this for *n* times.
3. **Random Swap (RS)**: Choose two words in the sentence randomly and swap their places in the sentence. Repeat this for *n* times.
4. **Random Deletion (RD)**: Remove each word randomly in the sentence with probability *p*.

EDA has shown good results in generating different permutations of questions using the input the question, however, the resultant sentences often do not have correct grammatical and semantic form which is not entirely useful.

2.6 Word-sense Disambiguation

Word-sense disambiguation (WSD) is still an open problem in the world of natural language processing which is defined as the task to determine which meaning of the word is activated by the usage of the word in a particular context based on the surrounding words. This task is challenging because we understand that words can have various meanings based on the context of the word's usage in the sentence. Therefore, the words are ambiguous because many words can be interpreted in different ways depending upon the context of their occurrence.

Raganoto et al. [20] categorised WSD approaches into two main categories namely, supervised approach and knowledge-based approach. The main difference between supervised and knowledge-based WSD approaches is that supervised WSD approach requires training of the sense-annotated corpus while knowledge-based WSD approach does not require any corpus training. One similarity between both approaches is that they rely on a database that is widely used for WSD called WordNet database [21] which is an inventory of word senses of words. Therefore, word senses of a given word can be found in the WordNet database. WordNet categorise English words of nouns, verbs, adjectives and adverbs into sets of cognitive synonyms known as synsets to each express a unique concept. Synsets are connected with one another via conceptual-semantic and lexical relations. Each synset consists of a list of words embodying the same sense. Therefore, if the WSD approaches are able to identify the correct sense of the word or phrase in a sentence, we can access WordNet to obtain the synset of the word and substitute the word or phrase with other words or phrases from the synset to generate new sentences.

The Michael Lesk Algorithm [22] is widely used for WSD which uses WordNet to gather the gloss of all the senses of the word in the sentence and then computes the maximum overlap with the senses returning whichever that generates the maximum overlap. For this project, an adapted Lesk algorithm is used to find the sense which is best related to the sentence excluding the stop words. A constant *k* is defined to limit the context of the matching sentence. The overlap is computed using the gloss of all the context words such that the most matching sense of the word will be chosen as the best sense of the word in the context.

Chapter 3 Methodology and System Architecture

The proposed solution (QG system) is a question generation engine consisting of two main components: a question generator utilising a combination of existing machine translation applications and another question generator that uses the Chinese language translated input English question to generate different permutations. The whole system is coded in the Python programming language with extensive applications of Python libraries and modules which will be explained and discussed in detail in the upcoming sections.

3.1 System Architecture

The overall system architecture for the proposed solution (QG system) is shown in Figure 3-1. For the architecture, we specifically adopted the data flow architecture design style because it allows us to divide and enclose the independent functionalities in individual components so that each component can be reused and improve the ease of maintainability.

This high-level data processing process consists of two main stages which will be described below:

1. **Combination of machine translation applications question generator.** Given an input question, this question generator will generate permutations of questions by translating the input question utilising a combination of machine translation applications into Chinese language and translating the sentence back to English language.
2. **Permutation of translated Chinese sentence question generator**. Given an input question, this question generator will generate permutations of questions by permutation of the Chinese translated question and translating the different permutations back to English language.

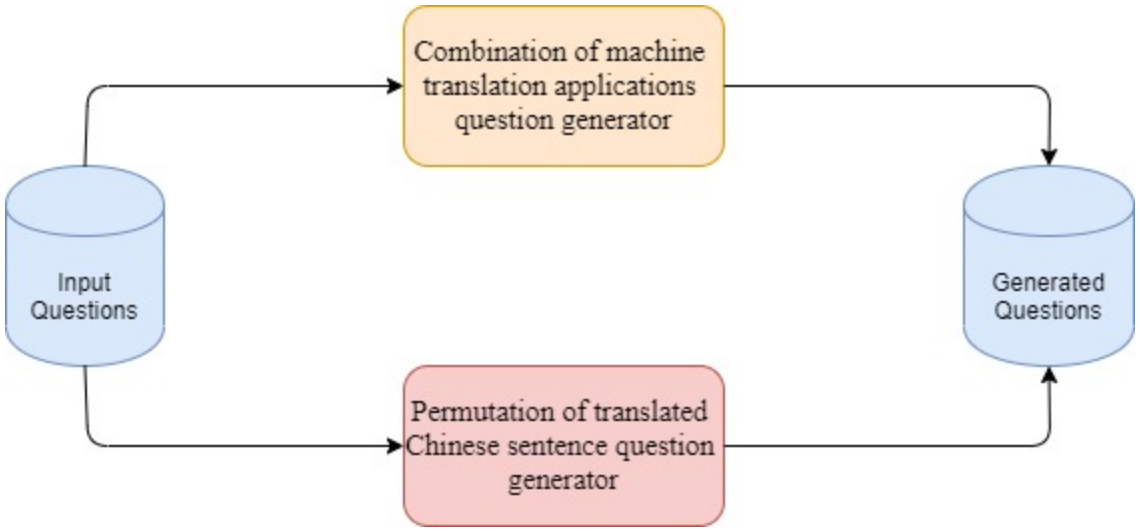

Figure 3-1: System architecture overview of QG System

3.2 Key Concepts

In this section, we will review some key concepts of text processing techniques that will be used in our QG system.

3.2.1 Tokenisation

Tokenisation refers to the process of separating a sentence into a string of words and punctuations. Each individual word or punctuation is taken as a *token*. Depending on the tokenisation requirements of the specific language processing application, punctuations or white spaces may be discarded and only output a sequence of string of words. However, for certain words, separating the punctuation and word results in the loss of meaning of the word as a whole with the punctuation, for example, words such as *"e.g."*, *"U.K."* and *"a.m."*. For these words, the full stop does not separate two words but rather have a meaning to it. In this case, statistical tokenisation is used in NLP programming libraries such as spaCy which uses statistical dependency parsing (discussed further in section 3.2.3) to determine the word and punctuation boundaries.

3.2.2 Part-of-speech Tagging

Part-of-speech (POS) tagging refers to the process of tagging each word in a sentence, after tokenising the sentence, with a corresponding part of speech tag based on the context the word is in the sentence. POS

tagging can be a difficult task as similar words can have different POS tags due to the word being in different contexts. There are several approaches for the process of POS tagging as described below:

1. **Lexical-based approach.** For this approach, each word in the sentence is assigned the POS tag that occurs the most frequently based on the training corpus.
2. **Rule-based approach.** For this approach, the word is assigned the POS tag where the rules dictate it to be. For example, there are rules such as all words that end with *-ing* must be a verb. This approach is applicable and effective when used with the lexical-based approach and the word that needs to be tagged does not appear in the training corpus, hence, the tagging rules will be applicable to the word to be tagged.
3. **Probabilistic-based approach.** For this approach, the POS tag that is assigned to the word which is based on the probability of the occurrence of a particular tag sequence. This approach requires several models to calculate and compute a table of probabilities of the occurrence of the tag or word sequences. One such model is the Hidden Markov Models (HMM).
4. **Deep learning approach.** For this approach, the use of RNNs is involved in the process of assigning POS tags to the words in a sentence.

The figure below will show the Penn Treebank POS tagset according to the Natural Language Toolkit (NLTK) package:

Number	**Tag**	**Description**
1	CC	Coordinating conjunction
2	CD	Cardinal number
3	DT	Determiner
4	EX	Existential there
5	FW	Foreign word
6	IN	Preposition or subordinating conjunction
7	JJ	Adjective
8	JJR	Adjective, comparative
9	JJS	Adjective, superlative
10	LS	List item marker

11	MD	Modal
12	NN	Noun, singular or mass
13	NNS	Noun, plural
14	NNP	Proper noun, singular
15	NNPS	Proper noun, plural
16	PDT	Predeterminer
17	POS	Possessive ending
18	PRP	Personal pronoun
19	PRP$	Possessive pronoun
20	RB	Adverb
21	RBR	Adverb, comparative
22	RBS	Adverb, superlative
23	RP	Particle
24	SYM	Symbol
25	TO	to
26	UH	Interjection
27	VB	Verb, base form
28	VBD	Verb, past tense
29	VBG	Verb, gerund or present participle
30	VBN	Verb, past participle
31	VBP	Verb, non-3rd person singular present
32	VBZ	Verb, 3rd person singular present
33	WDT	Wh-determiner
34	WP	Wh-pronoun
35	WP$	Possessive wh-pronoun
36	WRB	Wh-adverb

Figure 3-2: Penn Treebank POS tagset

3.2.3 Dependency Parsing

Dependency parsing refers to the process of parsing a sentence into a tree structure to represent the grammatical relations between “head” words which are the most grammatically important word in a sentence and “dependent” words which are the remaining words that change the head. There are a few common grammatical relations including adverbial modifier, nominal object and nominal subject. The purpose of dependency parsing is to verify the accuracy and precision of assignments of tokens after the extraction of tokens.

3.2.4 Machine Translation

Machine translation is a sub-field of computational linguistics that researches the utilisation of computer software to translate text or speech from one source language to another target language. There are several main approaches to machine translation which will be described below:

1. **Rule-Based Machine Translation (RBMT).** RBMT utilises the linguistic information about the source and target languages such as the morphological and syntactic rules and semantic analysis of both languages. The main idea involves connecting the structure of the source sentence with the structure of the target language using a parser and an analyser for the source language, a generator for the target language, and a transfer lexicon for the translation.
2. **Statistical Machine Translation (SMT).** SMT utilises the availability of corpuses of translated texts of a source language to a target language and bilingual texts. Source texts are first segmented into individual sentences and compared with the corpuses to find the statistically best representation of the source sentence in the target language.
3. **Neural Machine Translation (NMT).** NMT utilises artificial neural networks used for machine learning tasks to compute and predict the probability of the sentence sequences to find the best representation of the source sentence in the target language. This approach requires less memory compared to other traditional machine translation approaches.

3.2.5 Evaluation metrics to evaluate sentences

To evaluate the effectiveness of the proposed solutions to generate semantically similar permutations of questions, there are several metrics that will be used. These evaluation metrics which will be used will be described more below:

1. **Grammaticality correctness**. The grammaticality correctness that will be used in this report is known as the Flesch Reading Ease score [23]. This metric assesses the readability of a sentence or a paragraph of texts. There is no maximum or minimum score (negative scores are still valid) for this metric. The higher the output score, the easier and more readable the text is due to lower grammar errors, hence, representing that the text is grammatically sound and fluent. On the other hand, the lower the output score, the more confusing and incomprehensible the text is due to many grammar errors, hence, representing that the text is not grammatically sound and fluent.
2. **Lexical diversity**. Cosine similarity is used in texts to measure the lexical diversity or similarity between two input texts and outputs a resultant score that ranges from "0" to "1". This is done by converting the sentences into vector representations and measuring the cosine angle between the vectors to output the cosine similarity score [24]. Having a maximum score of "1" represents that the two input text is completely identical and are duplicates of one and another. Having a minimum score of "0" represents that two input texts are completely different.
3. **Semantic Similarity/ Meaning Preservation**. Semantic similarity is used to measure the preservation of meaning in the second sentence compared to the first sentence [25]. The sentences are first broken down into their individual tokens which will be assigned a POS tag to it. Words will be stemmed and using word sense disambiguation to find the most appropriate sense for every word in the sentence. The resultant similarity score of the sentences will be computed based on the similarity of the pairs of words. The similarity score ranges from "0" to "0.5". The higher the score, the higher the similarity in meaning between the two sentences.

3.3 Tools and Technologies

This section will discuss the tools, technologies and third party libraries used during the development of the QG system. Figure 3-3 provides a summary of tools and technologies used.

Tool/ Library	**Purpose**
Python 3.6	Main programming language
Jupyter Notebook	Interactive environment for efficient prototyping and data processing

Scikit-learn	Conversion of sentences to vector representations
NLTK	Synonyms lookup on WordNet
spaCy	NLP toolkit
Existing Machine Translation Applications	Translate sentences from a source language to target language
jieba	Segment Chinese sentences into their smallest most meaningful phrases and words

Figure 3-3: Summary of tools and libraries used

3.3.1 Python

Python[1] is the main programming language used in the development of the QG system due to its easy to understand language construct and having a wide range of third-party libraries for many development purposes. Python 3.6 was specifically chosen because of the freshly introduced features such as formatted string literals which helps to improve the efficiency of development.

3.3.2 Jupyter Notebook

Jupyter Notebook[2] is an interactive programming environment which allows users to create new programming files that consists of live code, interactive widgets and plots, etc. Powered together with the IPython Kernel[3], users are able to swiftly test out a component of their Python codes interactively as compared to traditional Python files where the whole file has to be implemented before knowing that the code is running well. This reduces the time in running the whole program and allows users to better know which part of their code has errors and allows them to rectify the errors earlier. It also supports graph plotting within the notebook with plotting libraries such as Matplotlib, making it a powerful tool for data processing and visualisation.

[1] https://www.python.org/
[2] https://jupyter.org/
[3] https://github.com/ipython/ipython

3.3.3 Scikit-learn

Scikit-learn[4] is an open-source and free software machine learning library for the Python programming language. It features various machine learning related algorithms such as classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, *k*-means and DBSCAN. For this project, scikit-learn was used to convert sentences into vector representations and using cosine similarity to measure the lexical diversity between sentences.

3.3.4 NLTK

Natural Language Toolkit (NLTK)[5] is an open-source and free natural language processing library made for building Python programs. For this project, NLTK was used to access the interface of WordNet to look up synonyms of words.

3.3.5 spaCy

spaCy[6] is an open-source high level natural language processing tool for the Python programming language. It is designed to handle large amounts of text data. For this project, spaCy was used to handle tokenisation, POS tagging and dependency parsing.

3.3.6 Existing Machine Translation Applications

There are many existing machine translation applications being offered by large search engine companies online for free translation services to translate a sentence from a source language to a target language. These translation tasks are usually done efficiently to produce results that greatly capture and maintain the semantic meaning from the source sentence in the target language.

As mentioned previously, there are many machine translation applications offered by large search engine companies such as *Baidu, Google, Microsoft Bing, Microsoft Neural, NetEase, Sogou, Tencent and Youdao*. A comparison in performance between these mentioned companies has been done by F. Yiqin [26]. In the experiment, excerpts of President Xi Jinping's speech at the 2018 Bo'Ao Forum were used as input texts to test the translation performance of these various machine translation applications when the excerpts were

[4] https://scikit-learn.org/stable/
[5] https://www.nltk.org/
[6] https://spacy.io

translated to English language. The English text output generated by the machine translation applications were compared against the official translation of the speech to evaluate the accuracy of the machine translation applications in being able to preserve the semantic meaning of the input Chinese speech.

The result of the experiment showed that *Google* ranked the highest in preserving the semantic meaning of the Chinese speech in the translated English output while *Microsoft Bing* ranked the lowest amongst the compared machine translation applications. The experiment also showed that as different machine translation applications use different training corpuses with varying sentence structures, translated sentences may have different sentence structures while preserving the semantic meaning of the original input text.

3.3.7 jieba

jieba[7] is an open-sourced, free and lightweight library that allows users to segment Chinese sentences into the smallest meaningful phrases and words. This is possible as jieba has their own dictionary of all possible Chinese phrases and includes the frequency of how likely the phrase will occur, hence, able to segment Chinese sentences into phrases and words effectively. It also allows users to add in their own dictionary of phrases so that the library is able to identify the phrases instead of segmenting the phrases into even smaller phrases or words. For this project, jieba is used to segment Chinese sentences so that we can permute the segments of phrases and words.

[7] https://github.com/fxsjy/jieba

Chapter 4 Implementation and Results

This chapter will introduce the implementation details of the proposed methods for the generation of semantically similar permutations of questions.

4.1 Machine Translation Applications

The use of machine translation was proposed as a method to generate semantically similar permutations of questions in this project. The motivation behind this method was to make use of the existing translation applications to translate a given English question to Chinese language and then translate the question in Chinese language back to English language. The aims of the method were trying to preserve the semantic meaning as well as to generate different permutations of the original input question through multiple translations. Another aim of this experiment was to observe if this method can retain the Chinese influence in the generated English questions to suit the style of English sentences spoken by the majority of Singaporeans.

4.1.1 Combining Existing Machine Translation Applications

As mentioned previously in section 3.3.6, an experiment conducted by a researcher concluded that the translation done by *Google* translation application produces the best results while *Microsoft Bing* ranked last with the worst translation results amongst the various existing machine translation applications that were compared. Using this information, we would like to propose using machine translation applications of varying performance to generate different permutations of English questions. This is because if we only choose the machine translation applications of the top few performances, the translated output questions would be rather similar hence, there would not be any significant difference in the permutations of the output questions.

The proposed system built will take the MSF Baby Bonus FAQ dataset as input and feed question by question into the various machine translation applications to translate those questions into Chinese language. The resulting Chinese question translated by each machine translation application will then be

cross fed into the various machine translation applications to generate English questions of different permutations as they are translated by different machine translation applications.

For this proposed system, we have selected a few machine translation applications namely, *Google*, *Microsoft Bing* and *Baidu*. These different machine translation applications will be used in the following combinations shown in the figure below to produce different permutations of English questions:

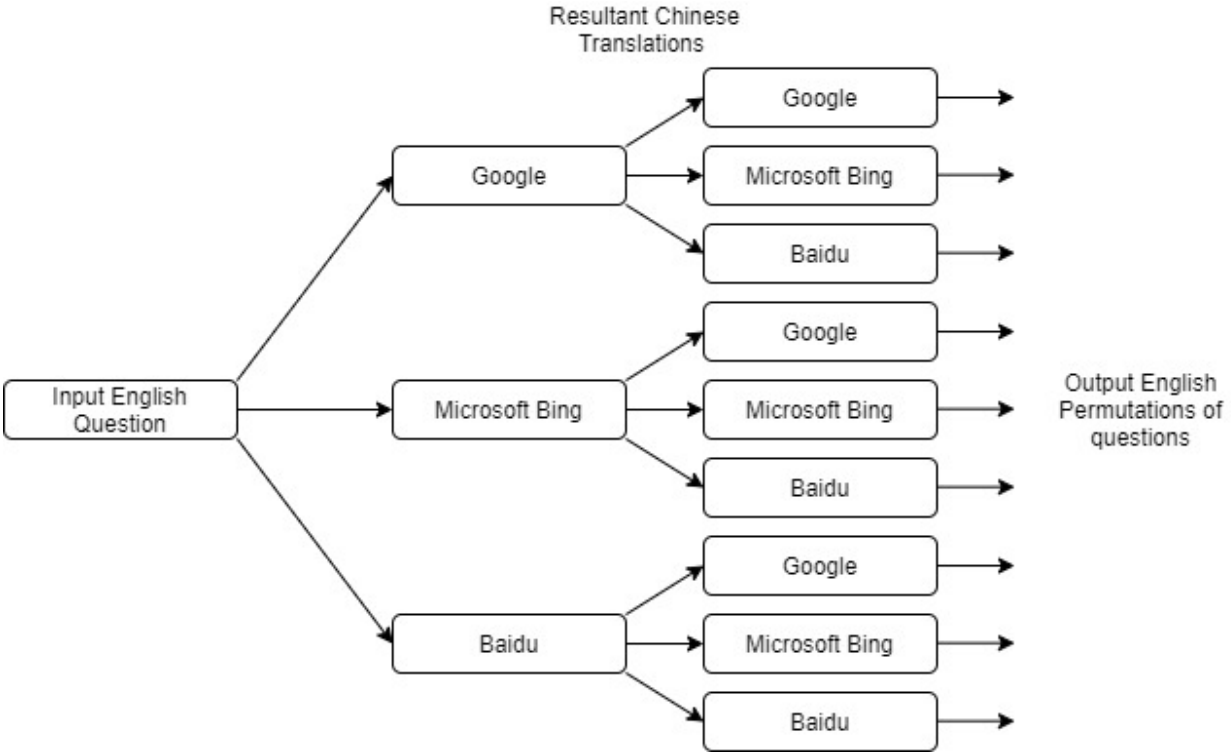

Figure 4-1: Flowchart of combinations of machine translation applications

Using the system built according to the flowchart shown above, we can see that, one input English question is capable of producing $3^2 = 9$ possible permutations as 3 machine translation applications were used. Hence, if we use more different machine translation applications, we can achieve more possible permutations. Even though using more translation machine applications can achieve more possible permutations, the resultant English questions might not have too much variance as the performance of some machine translation applications are rather similar and might even have duplicate results hence, not many permutations are being generated.

4.1.2 Results of combination of machine translation applications

For an example, we have chosen an English question with presence of Chinese language influence that will be spoken by most Singaporeans and observe how the results will turn out. The English question chosen, *"Can apply for the Baby Bonus Scheme before my baby is born?"*, follows the format of how Chinese sentences are formed which is the ***S** (Sentence) = **T** (Theme) + **R** (Rheme)* format as mentioned above in section 2.1.1. The front part of the question, "*Can apply for the Baby Bonus Scheme*", forms the topic or theme of the sentence and the rear part of the question, "*before my baby is born?*", forms the comment or rheme that is related to the topic of the question. The results produced using the proposed system are shown below:

Original English Question: Can apply for the Baby Bonus Scheme before my baby is born?	
Google to Google	Can my child apply for the baby bonus program before birth?
Google to Bing	Can I apply for a baby bonus plan before my child is born?
Google to Baidu	Can I apply for a baby bonus plan before my baby is born?
Bing to Google	Can I apply for a baby bonus plan before my baby is born?
Bing to Bing	Can I apply for a baby bonus plan before I give birth?
Bing to Baidu	Can I apply for a baby bonus plan before my baby is born?
Baidu to Google	Can I apply for the Baby Rewards program before my child is born?
Baidu to Bing	Can I apply for a baby incentive program before my child is born?
Baidu to Baidu	Can I apply for a baby award program before my baby is born?

Figure 4-2: Table of resulting permutations using all the machine translation applications

From the results, it can be observed that even though machine translation applications of varying performance were used, there are still a few duplicate permutations of English questions being produced. Among the 9 total resultant English questions generated, there were 7 unique permutations of the question, *"Can apply for the Baby Bonus Scheme before my baby is born?"*, being generated. This ratio shows a high percentage of unique permutations among all the permutations generated using this proposed system that utilises machine translation applications of varying performance to generate unique permutations of the

input English questions. The combinations of machine translation applications that generate unique permutations are shown in the figure below:

Original English Question: Can apply for the Baby Bonus Scheme before my baby is born?	
Google to Google	Can my child apply for the baby bonus program before birth?
Google to Bing	Can I apply for a baby bonus plan before my child is born?
Google to Baidu	Can I apply for a baby bonus plan before my baby is born?
Bing to Bing	Can I apply for a baby bonus plan before I give birth?
Baidu to Google	Can I apply for the Baby Rewards program before my child is born?
Baidu to Bing	Can I apply for a baby incentive program before my child is born?
Baidu to Baidu	Can I apply for a baby award program before my baby is born?

Figure 4-3: Table of resulting unique permutations using the proposed system

From the results produced, we can observe that the permutations were generated by creating synonyms for the words and phrases: *"baby", "Baby Bonus Scheme"* and *"before my baby is born"* when these phrases are translated into Chinese language and back to English language. The synonyms created during the process of translations will then replace the phrases in the input English question to form new permutations. We can also observe that the Chinese influence was retained in the resultant permutations of English questions similar to the original input question. Even though there was a high percentage of unique permutations generated, the extent of change in sentence structures and differences among all the permutations were largely limited to just the substitution of synonyms for the key phrases in the input English question.

The translation results generated by the proposed system were evaluated based on two metrics which are how grammatically correct the questions are and whether the semantic meaning of the question generated is preserved compared to the original input question. In order to evaluate a sentence based on the grammar, the sentence is further evaluated based on their fluency and comprehensibility. Based on the calculations, there will be an output score. This output score will have no maximum score and no minimum score. The higher the score simply means that the sentence is more readable with fewer grammatical errors while the lower the score means that the sentence is highly incomprehensible with bad grammar structure.

Besides generating permutations of semantically similar questions, another aim of this project is to attempt to generate questions with Chinese language influence. Due to the differences in sentence and grammar structure between English and Chinese sentences, a well formed Chinese sentence when directly translated into English sentence to preserve the structure of Chinese sentences and Chinese influence will tend to have a low grammaticality score that is calculated based on formal English sentences. Therefore, if a sentence or question translated from Chinese language as proposed in this system has a low grammaticality score, the sentence or question will tend to have a higher Chinese language influence. The grammaticality score of each of the questions generated by the system using the example question is shown in the figure below:

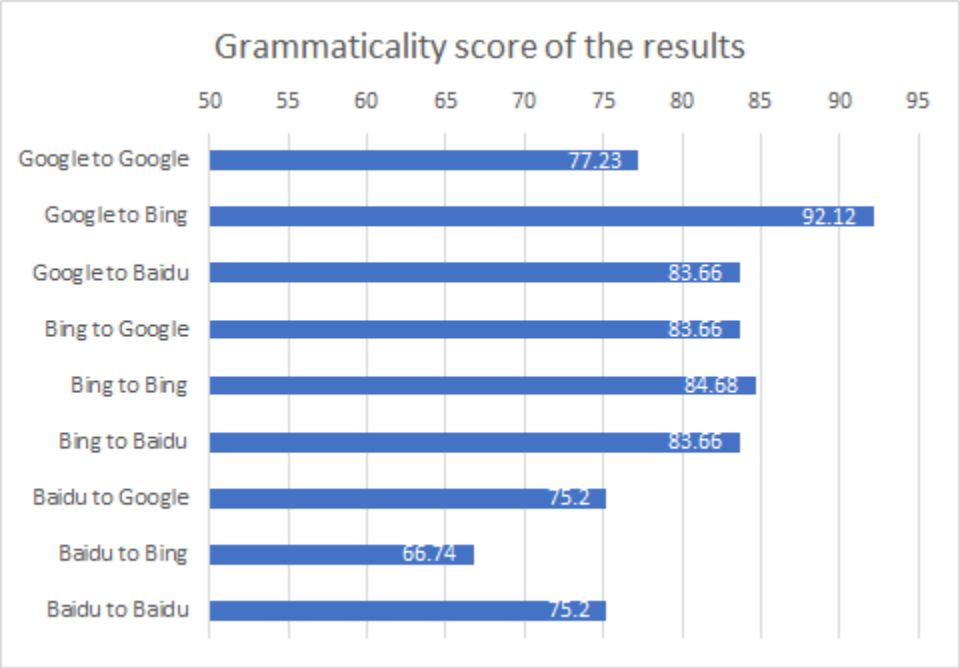

Figure 4-4: Grammaticality score of the results

It is observed that the combination of *Google* and *Microsoft Bing* machine translation applications when *Google* was used to translate the input English question to Chinese language and then *Microsoft Bing* was used to translate the intermediate question in Chinese language back to English language, this combination generates sentence to have the best grammar score (92.12) compared among the different permutations generated. As mentioned earlier, these grammar scores are calculated based on formal English language so having low grammar scores are acceptable as those sentences generated with low grammar scores have more Chinese influence in them hence, resulting in the sentence to have a low grammar score. It is also

observed that the combination of *Baidu* and *Microsoft Bing* machine translation applications, this combination generates sentences to have the worst grammar score (66.74) compared among the different permutations generated. This is further evident from the results in Figure 4-2 that the resultant question generated by the *Baidu-Microsoft Bing* combination is understandable with Chinese influence found in the question.

In order to measure the preservation of semantic meaning of the original question, the resultant questions generated by the proposed system are being compared to the input English question. For each comparison, a semantic meaning score will be calculated and output a score with maximum value of 0.5 and minimum value of 0. A maximum score will mean that the semantic meaning of the original sentence is preserved in the generated sentence with both sentences delivering the same message while a minimum score will mean that the meaning of the original sentence is lost in the generated sentence and has a totally different meaning compared to the original question. The semantic meaning preservation score of each of the questions generated by the system using the example question is shown in the figure below:

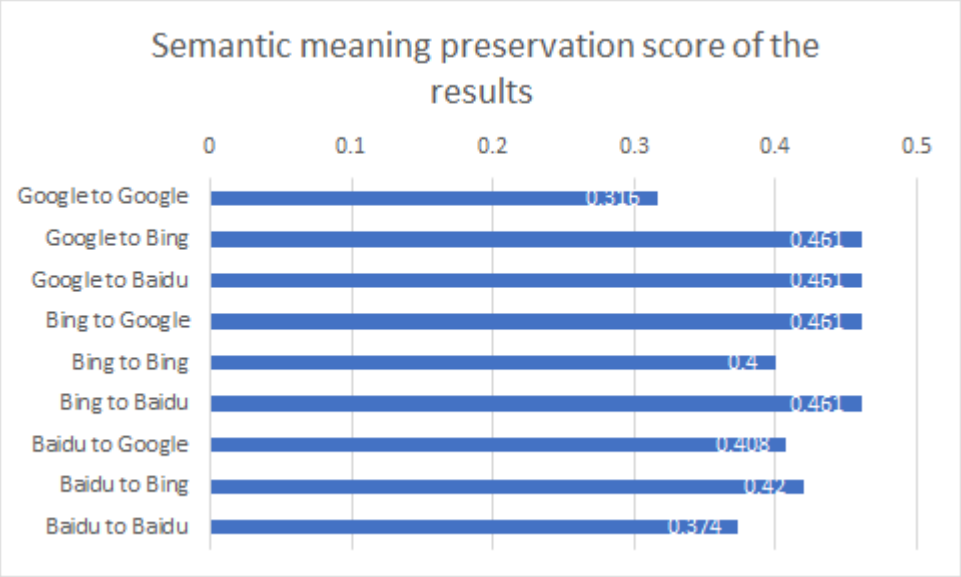

Figure 4-5: Semantic meaning preservation score of the results

It is observed that the semantic meaning preservation score of the results are all relatively high showing that the semantic meaning of the input English question is preserved in the generated sentences.

Based on the evaluation as a whole, the most ideal situation for the generated English questions using a combination of machine translation applications is for them to achieve a low grammaticality score and high on the semantic meaning preservation score. This is so that the objectives of this project are met where the Chinese influence is retained in the generated questions while preserving the semantic meaning of the original input question.

4.2 Direct translation from Chinese sentences

As discussed earlier, one of the aims of this project is to generate permutations of questions with Chinese language influence so that the sentence generated will resemble the sentences that Singaporeans will speak typically. One way to preserve the Chinese influence in English sentences is to directly translate Chinese sentences to English sentences. Using this method, the sentence structure of the Chinese sentence can be retained in the translated English sentence.

4.2.1 Permutation of translated Chinese sentences

In order to fulfill another aim of generating different permutations of English questions, we make use of the sentence structure of Chinese sentences to generate different permutations. Similar to English language, Chinese sentences are composed of phrases and words. We make use of this point and translate the original input English question into Chinese language so that we can take advantage of the structure of Chinese sentences and create permutations. Then, we segment the translated Chinese sentences into their smallest meaningful phrases and words. We have chosen an example English question from the MSF Baby Bonus FAQ dataset, *"How is the birth order determined?"*, and translated the question to Chinese language to demonstrate the segmentation of Chinese sentences. The segmentation of Chinese sentences is achieved by using the Python library called "jieba" which was mentioned in section 3.3.7. The segmented Chinese question is shown in the figure below:

Original English Question: How is the birth order determined?				
Translated Chinese Question：如何确定出生顺序**?**				
Segmented Chinese Sentence into phrases:	如何	确定	出生	顺序
Meaning of each Chinese phrase in English:	How to	determine	birth	order

Figure 4-6: Segmented Chinese question with their meaning

As shown in Figure 4-6 above, the Chinese sentence translated from the original English sentence can be segmented into 4 phrases. After segmenting the translated question in Chinese language into their smallest meaningful words and phrases, we realise that we can permute how each of the segments of the Chinese question can be joined with the previous and following segment to form larger Chinese phrases which can be translated into different formats of English phrases. An example of the permutation of Chinese phrases segmented from a sentence using the translation of the English question, "*How is the birth order determined?*", will be shown below:

Original English Question: How is the birth order determined?				
Translated Chinese Question：如何确定出生顺序**?**				
Segmented Chinese Sentence into phrases:	如何	确定	出生	顺序
Different permutations:				
1.	如何确定出生顺序			
2.	如何确定出生			顺序
3.	如何确定		出生顺序	

4.	如何确定		出生	顺序
5.	如何	确定出生顺序		
6.	如何	确定出生		顺序
7.	如何	确定	出生顺序	
8.	如何	确定	出生	顺序

Figure 4-7: Different permutations of segmented Chinese phrases

As shown in Figure 4-7 above, the segmented Chinese sentence consisting of 4 different phrases can be permuted to form different permutations of how each of the phrases are connected to the phrase before and after it to form larger phrases so that each phrase can be translated into different phrases in English. In the example shown above, a Chinese sentence of 4 phrases can be permuted to generate 8 different permutations of how these Chinese phrases are connected.

In order to generate different such permutations, we have to manipulate the spaces in between each phrase. For a Chinese sentence that has 4 phrases as shown in the example above, there would be 3 possible spaces throughout the sentence. We would assign binary values to each individual space in between the phrases. A value of *"1"* would be assigned to represent the presence of a space and a value of *"0"* would be assigned to represent the absence of a space which means that the Chinese phrases connected originally by a space would join together to form a larger Chinese phrase. All the permutations will be generated by considering all possible binary permutations of the spaces. Using the case of the Chinese sentence shown in Figure 4-7 above, after segmenting the Chinese sentence, the result will consist of 4 phrases. For a Chinese sentence that consists of 4 phrases, there will be 3 spaces separating the 4 phrases. Hence, the total number of all binary combinations of the 3 spaces will be equal to $2^3 = 8$ permutations. Therefore, for a Chinese sentence that consists of *"N"* number of Chinese phrases, there will be *"N – 1"* number of possible spaces in between the *"N"* number of phrases. Considering the number of spaces, there can be a total of 2^{N-1} number of possible permutations of Chinese sentences.

After forming the different permutations consisting of segmented Chinese phrases, we made use of machine translation applications to translate each segment of Chinese phrase back to English language. Using the translated English phrases from the different segments of Chinese phrases, we can then concatenate these

English phrases together to form new English sentences or questions that were translated and permuted from the input English question. This was done so that the English sentences or questions generated will retain the sentence structure of Chinese sentences and hence, introduce Chinese influence into the generated English sentences. The figure below will show the results after translating each permutation of Chinese phrases into English language and how the English sentence would look like after concatenating the English phrases:

	Chinese sentences with different permutations of segmentations:	Concatenation of English phrases after translated from each segmented Chinese phrase:
1.	如何确定出生顺序	How to determine the birth order
2.	如何确定出生/ 顺序	How to determine birth/ order
3.	如何确定/ 出生顺序	How to determine/ Birth order
4.	如何确定/ 出生/ 顺序	How to determine/ Born/ order
5.	如何/ 确定出生顺序	how is it/ Determine birth order
6.	如何/ 确定出生/ 顺序	how is it/ Determined to be born/ order
7.	如何/ 确定/ 出生顺序	how is it/ determine/ Birth order
8.	如何/ 确定/ 出生/ 顺序	how is it/ determine/ Born/ order

Figure 4-8: Concatenation of English phrases after translated from each segmented Chinese phrase

From Figure 4-8 above, we can see the concatenation of English phrases after translating from each segmented Chinese phrase. The slash, "/", character segments the Chinese sentences into different permutations of Chinese phrases. We can see the direct translation of each segmented Chinese phrase in the corresponding row of English sentences which is also segmented by the slash character. The Chinese language influence is also apparent in these generated sentences as it continues to follow the Chinese sentence structure as mentioned in section 2.1.1. An example will be shown below (Figure 4-9) using the generated sentences from Figure 4-8:

Original English Question: How is the birth order determined?		
Chinese sentence structure: **S** (Sentence) = **T** (Theme) + **R** (Rheme)		
	T (Theme)	**R** (Rheme)
1.	How to determine	the birth order
2.	How to determine	birth order
3.	How to determine	Birth order
4.	How to determine	Born order
5.	how is it Determine	birth order
6.	how is it Determined	to be born order
7.	how is it determine	Birth order
8.	how is it determine	Born order

Figure 4-9: Resulting permutations with Chinese influence

As shown in Figure 4-9, the generated sentences follow the Chinese sentence structure consisting of a Theme which is the topic of the sentence and Rheme which is a comment that is related to the topic of the sentence. We can observe that the front segment of the generated sentences corresponds to the Theme of the sentence and the rear segment corresponds to the Rheme of the sentence.

4.2.2 Analysis of translated English sentences from Chinese Phrases

After observing the results from Figure 4-8, we realise that there are also duplicate sentences generated. A proposed method was to introduce a metric to measure the similarity between the sentences to identify the duplicate sentences. To identify duplicate sentences, cosine similarity metric mentioned in section 3.2.5 was used to find sentences that are identical. For duplicate sentences, their cosine similarity would be equal to 1 and therefore, by keeping track of those pair of sentences that have a cosine similarity of 1, we would be able to identify the duplicate sentences. To achieve this, all generated sentences would undergo similarity testing against all other generated sentences pair by pair to find the duplicate sentences. After identifying the duplicate sentences, we would discard them and only retain the unique permutations of sentences. This

was done so that we only generate unique permutations of sentences to train the question answering model for the FAQ chatbot for the MSF Baby Bonus Scheme.

To further make use of the cosine similarity to measure the similarity between the sentences, we used cosine similarity to identify the generated sentences that were the most dissimilar to the original input English question. This was done so that we can identify the permutation of sentences that has the greatest extent of change compared to the input English question. Therefore, we would be able to retain the generated sentences that are most different and use these different permutations of sentences to train the question answering model for the FAQ chatbot more effectively so that the question answering model will have a more diverse dataset of questions to train. The cosine similarity of each generated sentence compared to the example input English question, "*How is the birth order determined?*", will be shown in Figure 4-9 below:

Original English Question: How is the birth order determined?		
	Concatenation of English phrases after translated from each segmented Chinese phrase:	Cosine Similarity compared to the original input English question:
1.	How to determine the birth order	0.667
2.	How to determine birth order	0.548
3.	How to determine Birth order	0.548
4.	How to determine Born order	0.365
5.	how is it Determine birth order	0.667
6.	how is it Determined to be born order	0.577
7.	how is it determine Birth order	0.667
8.	how is it determine Born order	0.500

Figure 4-10: Cosine similarity comparing the generated question with input question

From Figure 4-10 above, we can observe the cosine similarity between the generated sentence and the original input English question and realise that sentence number 4, *"How to determine Born order",* has the lowest cosine similarity score (0.365) when compared to the original input question. The cosine

similarity scores of other generated sentences are also rather low showing that this proposed method is capable of generating sentences that have a good extent of change compared to the input English question.

For this proposed method, the measure of how grammatically correct the generated sentence is not of a high priority as direct translation of each segment of Chinese phrase to English language was used to generate the English sentence. Hence, there will be a high chance that most sentences will not be grammatically sound and instead would have Chinese influence introduced. Therefore, the grammaticality score of these generated sentences would not be important.

4.3 Discussion of Results of Proposed Methods

In a comparison between the two proposed solutions to generate semantically similar permutations of an input English question with Chinese influence, the second approach that utilises the direct translation of segmented Chinese phrases to English language performs better than the first approach to generate sentences using a combination of machine translation applications.

The first approach produces English sentences by translating the original input English question to Chinese language then translates the Chinese sentences back to English language. This method is able to produce various permutations through the utilisation of the different combinations of machine translation applications. The generated sentences manage to preserve the semantic meaning of the input English question due to the already highly trained translation models used in these machine translation applications. One of the limitations is due to the fact that the entire Chinese sentence is translated to English language and since the translation models are highly trained, the English sentence generated will be a grammatically sound English sentence and have high grammaticality scores. This would therefore mean that the generated English sentences would have a weak but still noticeable Chinese influence in them as we mentioned previously that usually English sentences with Chinese influence would not have good grammaticality scores. Another limitation would be the number of sentences that can be generated. Using the proposed approach, the number of sentences generated would be limited by the number of machine translation applications used as if there are *"N"* number of machine translation applications used, there would only be *"N^2"* number of sentences generated. Increasing the number of machine translation applications to be used might not be effective as there might be a possibility that the generated sentences would be too similar as

the performance of some machine translation applications are quite similar. Therefore, this would result in more duplicate sentences to be generated and hence wasting the computer resources and memory.

On the other hand, the second approach generates English sentences which have more Chinese influence introduced into them as the generated English sentence is formed by concatenating the direct translation of each segmented Chinese phrase or word, hence retaining the Chinese sentence structure when translated back to English language. In addition to being able to retain the Chinese influence in the generated English sentences, the proposed method is capable of generating more permutations of the English question if the input English question is longer. A longer English question will mean that the translated Chinese question will be longer and will therefore be made up of more phrases and words after segmentation of the Chinese question. Having made up of more phrases and words, there can be more permutations of concatenating the phrases and words to form different lengths of phrases to be translated back to English language, hence, generating more permutations of English questions if the original English question is long and would not be limited as compared to the first approach where the number of English questions generated is limited by the number of machine translation applications are used in combination. One possible limitation would be that shorter input English questions might result in shorter translated Chinese sentences so when the Chinese sentence is segmented, there are fewer permutations generated. For example, if the translated Chinese sentence only has 3 phrases or words after segmentation, the number of sentences generated will be $2^{3-1} = 2^2 = 4$ as given by the formula in section 4.2.1. This number of generated sentences might not be enough to train the question answering model for the FAQ chatbot.

Comparing both proposed approaches, both approaches are able to generate permutations of semantically similar questions with some Chinese influence introduced into the generated sentences. However, the second approach is more effective in generating more permutations of questions using the input English question and introducing a stronger Chinese influence into the generated sentences compared to the first approach.

Chapter 5 Conclusion

In this thesis, we introduced the **QG system** and the linguistic differences between the English language and Chinese language which made use of it to generate permutations of semantically similar questions with Chinese influence with the goal to augment the limited training set of queries. This is done so as to increase the limited training set of queries to train the automated FAQ answering system to understand more questions so that the best answer can be selected from the database to answer the query of users.

Two main approaches were proposed to generate permutations of semantically similar questions with Chinese influence. The first approach uses the combination of machine translation applications and the second approach uses the permutation of Chinese phrases and words after segmenting the translated Chinese sentence. From the experiment done to compare both approaches, we conclude that both approaches are able to generate permutations of semantically similar questions with some Chinese influence. However, the second approach is more effective in generating more permutations of questions using the input English question and introducing a stronger Chinese influence into the generated questions compared to the first approach.

5.1 Future Works

The approaches proposed in this thesis are not perfect and have their own limitations. Therefore, we would like to propose some future works which can be done by researchers interested in the same field of question generation for FAQ answering systems. The following pointers in this section will discuss a few directions and methods which can be explored to generate questions that resemble the way English sentences are spoken by Singaporeans.

5.1.1 Machine Translation

Machine translation for generation of permutations of questions may be further improved by training the models with parallel corpus in colloquial language mined from online blogging platforms such as Twitter and Weibo which has a large base of Chinese users with a huge database of casual sentences written by

Chinese users. This helps to train the model to generate English questions which have a lesser degree of formality in English [27].

5.1.2 Style Transfer

Style transfer is considered to be a new research topic and is commonly regarded as a machine translation task. This task requires a large corpus written in a source language and another target corpus of the same content and language but written in a totally different style. It is said that it is possible to train a machine translation model to "translate" a given input text in a language into a new piece of text in the same language but with the new style, hence, generating sentences with new permutations. However, this approach requires large amounts of training data with texts written in different styles for the model to train and extract the different writing styles. Currently, there are not many high quality corpora available to be used. If there are large and high quality corpus available, this can possibly be a better approach in generating more permutations of questions.

5.1.3 Structural Reconstruction

Common Chinese FAQ data sets can be used to train models to obtain the Chinese sentence structures for questions. Sentence reconstruction can then be done after extracting the sentence patterns and collocation pairs by substituting the synonym pairs and generating English questions with Chinese sentence structures. This would generate higher quality English questions with Chinese language influence introduced into the permutations generated.

5.1.4 Named Entity Linking and Understanding

Named entity linking and understanding can be introduced to capture the keywords and generate permutations of questions based on the identified keywords. For example, if we are able to identify and link the named entities in the question, *"Can you tell me more about the BBS?"*, we can generate questions related to the identified named entities like, *"What is Baby Bonus Scheme about?"*.

References

[1] A. Bolen, *What are Chatbots?* [Online] Available: https://www.sas.com/en_us/insights/articles /analytics/what-are-chatbots.html

[2] G. Cong, L. Wang, C.-Y. Lin, Y.-I. Song, and Y. Sun, "Finding question-answer pairs from online forums," *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 467–474, 2008.

[3] B. P. Yap, "Question Generation for FAQ Answering," 2019.

[4] K. A. Famili, "Generating semantically similar permutations of questions by clustering," 2017.

[5] S. D. o. Statistics, *Population trends, 2019,* 2019.

[6] P. Lee, "English most common home language in Singapore, bilingualism also up: Government survey", *The Straits Times*, 2016

[7] Q. Zhu, "Main influence of the Chinese language on English learners," *Journal of Language Teaching and Research*, vol. 1, no. 2, p. 668, 2010.

[8] N. Duan, D. Tang, P. Chen, and M. Zhou, "Question generation for question answering," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 866–874.

[9] V. Rus, Z. Cai, and A. C. Graesser, "Evaluation in natural language generation: The question generation task," in *Proceedings of the Workshop on Shared Tasks and Comparative Evaluation in Natural Language Generation*, 2007, pp. 20–21.

[10] Y. Li, N. Duan, B. Zhou, X. Chu, W. Ouyang, X. Wang, and M. Zhou, "Visual question generation as dual task of visual question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6116–6124.

[11] S. Reddy, D. Raghu, M. M. Khapra, and S. Joshi, "Generating natural language question-answer pairs from a knowledge graph using a rnn based question generation model," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, 2017, pp. 376–385.

[12] L. Pan, W. Lei, T.-S. Chua, and M.-Y. Kan, "Recent advances in neural question generation," *arXiv preprint arXiv:1905.08949*, 2019.

[13] H. Ali, Y. Chali, and S. A. Hasan, "Automation of question generation from sentences," in *Proceedings of QG2010: The Third Workshop on Question Generation*, 2010, pp. 58–67.

[14] I. Labutov, S. Basu, and L. Vanderwende, "Deep questions without deep understanding," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015, pp. 889–898.

[15] N. Duan, D. Tang, P. Chen, and M. Zhou, "Question generation for question answering," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 866–874.

[16] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[17] Y. Wang, J. Zheng, Q. Liu, Z. Zhao, J. Xiao and Y. Zhuang, "Weak Supervision Enhanced Generative Network for Question Generation," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)*, 2019.

[18] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1700–1709.

[19] J. W. Wei and K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," *arXiv preprint arXiv:1901.11196*, 2019.

[20] A. Raganato, J. Camacho-Collados, and R. Navigli, "Word sense disambiguation: A unified evaluation framework and empirical comparison," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, 2017, pp. 99–110.

[21] G. A. Miller, "Wordnet: A lexical database for english." Communications of the ACM, vol. 38, pp. 39–41, 1995.

[22] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone," in *Proceedings of the 5th Annual International Conference on Systems Documentation*, ser. SIGDOC '86. New York, NY, USA: ACM, 1986, pp. 24–26. [Online]. Available: http://doi.acm.org/10.1145/318723.318728

[23] A. Kher, S. Johnson and R. Griffith, "Readability Assessment of Online Patient Education Material on Congestive Heart Failure.," *Advances in preventive medicine*, 2017, 9780317. [Online]. Available: https://doi.org/10.1155/2017/9780317

[24] B. Li and L. Han, "Distance Weighted Cosine Similarity Measure for Text Classification," in *Intelligent Data Engineering and Automated Learning – IDEAL 2013*, 2013, pp. 611 – 618.

[25] N. D. Thanh and T. Simpson, "Measuring Similarity between sentences," 2018

[26] F. Yiqin, *Who offers the best chinese-english machine translation? A comparison of google, microsoft bing, baidu, tencent, sogou, and netease youdao.* [Online]. Available: https://yiqinfu.github.io/posts/machine-translation-chinese-english-june-2018/.

[27] Ling, Wang, Xiang, Guang, Dyer, Chris, Black, Alan, Trancoso, and Isabel, “Microblogs as parallel corpora,” in *Proceedings of the 51st Annual Meeting on Association for Computational Linguistics*, ser. ACL ’13, Sofia, Bulgaria: Association for Computational Linguistics, 2013.