# University of Miskolc

Faculty of Mechanical Engineering and Informatics

# Extended Sentence Parsing Method for Text-to-Semantic Application

Ph.D Dissertation Booklet

Author:

**Walelign Tewabe Sewunetie**
M.Sc in Computer Science

József Hatvany Doctoral School of Information Science, Engineering and Technology

Research Area
**Applied Computer Science**

Research Group
**Knowledge Base and Data Analysis**

Head of Doctoral School:
**Prof. Dr. Jenő SZIGETI**
Academic Supervisor:
**Prof. Dr. László KOVÁCS**

Miskolc, Hungary 2024

# Contents

Contents

# Chapter 1

# Background

The field of Natural Language Processing (NLP) has advanced significantly in recent years, resulting in the creation of complex systems that indicate a deep comprehension of human language in terms of both generation and comprehension. The expanding significance of NLP in various applications has led to focused research endeavors focused on enhancing language model capabilities, particularly in the field of text-to-semantic applications. With the increasing significance of language understanding in learning and information retrieval settings, Automatic Question Generation (AQG) has gained attention. AQG can accomplish more than two things: it makes it possible to create an automated assessment system from educational materials and improves information retrieval system performance. The dissertation aims to solve important issues and innovate in the field of sentence parsing, driven by the deep significance of AQG in educational and information retrieval contexts. The research is motivated by the need to expand and improve on current sentence parsing techniques, with a particular emphasis on their use in the larger context of text-to-semantic applications, as language comprehension becomes more and more important.

### 1.0.1 Objective and Research Question

The primary goal of this dissertation is to develop and evaluate extended sentence parsing methods that enhance the accuracy and efficiency of text-to-semantic applications, with a particular emphasis on AQG. To achieve this objective, the research seeks to answer the following key questions:

1. What are the limitations of existing sentence parsing methods in the context of AQG and text-to-semantic applications?

2. How can extended dependency parsing methods address the challenges faced in AQG?

3. What is the performance of MLP-based sentence parsing in comparison to traditional template-based approaches for AQG?
4. How can a hybrid parser, incorporating ChatGPT-based sentence parsing, contribute to semantic graph induction in text-to-semantic applications?

This dissertation aims to contribute novel insights and methods to the field of sentence parsing for text-to-semantic applications, focusing on AQG. The scope of the research encompasses the identification of limitations in current parsing methods, the development and evaluation of extended dependency parsing and multilayer perceptron-based models, and the exploration of a hybrid parser for utilizing ChatGPT-based techniques. The intended contributions of this research include the advancement of sentence parsing techniques that improve the accuracy of AQG, ontology creation, and semantic graph induction. By addressing the identified limitations and proposing innovative methods, this dissertation aims to enhance the overall efficiency of text-to-semantic applications.

### 1.0.2 Natural Language Processing and its Role

The dynamic field of NLP is focused on the relationship between human language and computers. It has evolved throughout time to become a cornerstone of AI, vital to a wide range of uses. NLP makes it possible for machines to understand, interpret, and produce language similar to humans, bridging the gap between computational systems and the many aspects of natural communication. NLP is important because of its many applications, which affect how we use technology and handle large volumes of textual data. NLP has influenced many aspects of our digital lives, including sentiment analysis, machine translation, and speech recognition, information extraction, question answering, and engaging in lengthy conversations with humans.

Complex sentence parsing, analysis, and semantic understanding are at the core of NLP. Investigating sentences into their constituent parts, interpreting syntactic patterns, and drawing conclusions from word choices are all part of parsing. Text comprehension is enhanced by this process, which is essential for understanding the meanings hidden in natural language. In the broader context of text-to-semantic applications, NLP plays a central role. The ability to parse sentences and extract semantic meaning is fundamental for tasks like AQG and semantic graph induction. These applications require a deep understanding of language structures and relationships between entities, which NLP endeavors to provide.

**Syntactic Analysis**: A fundamental aspect of NLP is syntactic analysis, which aims to understand every aspect of grammar and sentence structure. Syntactic analysis techniques parse the source text to identify grammatical structures and dependencies. By understanding the syntax of the text, the system can generate questions that maintain grammatical correctness and coherence with the source material. Understanding the placement of words in a sentence and their grammatical relationships depends heavily on this crucial component. Syntactic parser evolution: from classical rule-based techniques to modern probabilistic and neural network-based methodologies. Prior rule-based systems were limited in their ability to handle the complexity of natural language and relied on human-crafted grammatical rules and linguistic knowledge, which helped to provide the foundation for basic understanding.

**Semantic Understanding**: Simplified understanding is a key concept in NLP that overcomes language structure and focuses on revealing the meaning that is contained in words, phrases, and sentences. Simplified comprehension explores the essential meanings that words and constructs within a particular context convey, whereas syntax analysis focuses on extracting the meaning and semantics of the text. It involves understanding implicit meanings, details, and plain meanings that all add to the complexity of human communication. Simplified analysis addresses the clarity of meaning as opposed to syntax, which is concerned with the structure and order of words. Capturing the intended simplicity of a text can be difficult due to ambiguities, context-dependent interpretations, and language's dynamic character.

**Automatic Question Generation System**

AQG relies on NLP as a foundational element, offering indispensable tools for automated language comprehension. This section discusses the evolution of NLP, tracing its journey from historical roots to contemporary methodologies that shape AQG research. AQG, distinguished by its capacity to transform declarative statements into interrogative forms, facilitates a comprehensive exploration of the underlying material. NLP, situated at the intersection of artificial intelligence and linguistics, empowers computers to understand human language expressions.

AQG, defined as the generation of syntactically sound, semantically correct, and relevant questions from diverse input formats, hinges on technological advancements. The shift from manual to automated systems in education exemplifies this evolution, where traditional question generation by academicians has transformed. This section explores diverse techniques for question generation, emphasizing that the choice of technique depends on application requirements, source text quality, and desired question quality and diversity. Modern AQG systems often leverage a combination of techniques, incorporating NLP, ontology, and ML to enhance question relevance. The following discussion outlines common AQG techniques, showcasing the intersection of NLP with ontology and machine learning in question generation.

**Template-Based Method**: Template-based AQG techniques rely on predefined question templates that contain placeholders for specific information extracted from the source text. These templates serve as a structured framework for generating contextually relevant questions. For example, consider a template tailored to information about the capital of Ethiopia, such as: "What is the capital city of [country]?" In this template, the placeholder [country] can be filled in with the extracted information from the source text about Ethiopia. The system identifies the relevant content, such as "Ethiopia," and populates the template, resulting in a specific question like "What is the capital city of Ethiopia?" Template-based techniques offer a straightforward and systematic approach to question generation, making them especially useful for scenarios where specific types of questions need to be consistently generated from similar structures in the source text. These techniques can be adapted to various domains and information types, providing a flexible solution for automatically generating questions tailored to the content at hand.

Among various AQG methods [1], a template-based method is the oldest, and it can be easily implemented. Template-based method use templates containing fixed text, and some placeholders that are populated from the given content. According to the thesis report [2], the templates are created by focusing on the events (actions, happenings), and existents (characters, settings). In addition, I observed that most current templates ask about the subject, the predicate, and the object of the events and existents. In the development of the template-based question generation method, first, I need to prepare a quality and representative dataset. While, due to the complexity of natural language structure, it is very difficult to create general-purpose templates, [3]. In this study, I have developed an open-ended template-based system.

**Rule-Based Method**: Rule-based AQG techniques, employing dependency parsing, rely on predefined grammatical rules and templates to systematically generate questions. These rules provide instructions on how to extract or transform information from the source text into question forms. For instance, consider a rule that identifies sentences beginning with interrogative words like "Who," "What," or "Where" and dictates their transformation into questions. In the context of information related to Ethiopia, a rule could identify a sentence like "Ethiopia is known for its rich cultural heritage," and based on the rule, generate the question "What is Ethiopia known for?" The use of dependency parsing enhances the sophistication of rule-based techniques by considering the syntactic dependencies between words in a sentence. By understanding the grammatical relationships, these techniques can more accurately transform statements into questions. Rule-based approaches are valuable for maintaining grammatical correctness and generating questions that adhere to specific linguistic structures.

**Neural Network-based Method**: Researchers from other disciplines have recently become interested in the research topic of AQG for educational objectives. Cohen [4] proposed that the substance of a question can be represented as an open formula with one or more unbound variables in one of the first works on questions. While question generation research has been done for a long time, the use of AQG for educational purposes has attracted the attention of several academic communities in recent years. Questions have also been a major topic of study in computational linguistics where models of the transformation from answers to questions have also been developed. Previous studies have specifically addressed the generation of questions for educational objectives, as evidenced by Heilman et al[5], who showed that a combination of AQG and manual correction can be more time-efficient compared to solely manual authoring. Authors [6] created an automated reading tutor that uses AQG to help students improve their comprehension skills while reading a text.

The research conducted by Serban et al. [7] proposed employing a neural network approach to formulate factual questions based on structured data, as opposed to generating questions directly from textual content. Zhou et al. [8], in their study, conducted an initial exploration into question generation from text utilizing neural networks. They introduced the NQG framework, which enables the generation of natural language questions from text without relying on pre-defined criteria.

The advanced NLP techniques employed for the textual question creation include Natural Language Understanding (NLU) and Natural Language Generation (NLG). First, the system has to understand the input text which is NLU, and then it has to generate questions also in the form of text that is NLG. The article[9] presented a system for generating factual inquiries from unstructured material. They combine numerous ML algorithms with classical linguistic methodologies based on sentence patterns. In the disciplines of NLP and computer vision, generating natural language queries for picture understanding is a hot topic. Regarding the implementations of the learning modules the most dominant solution is neural network based architecture, specially the MLP and RNNs[10].

**Semantic Based Method**: The paradigm of text-to-semantic applications encompasses various methodologies aimed at extracting deeper meaning from textual content. This application goes beyond traditional question-generation approaches by incorporating semantic understanding to formulate questions that reflect a more profound grasp of the underlying meaning within the text. In the context of semantic-based AQG, the emphasis is on leveraging advanced language understanding to extract not only syntactic structures but also the semantic details present in the text. This approach aims to generate contextually relevant questions and align with the deeper meaning embedded in the content.

Within the semantic-based framework, AQG goes beyond conventional syntactic analyses. It leverages advanced language understanding, including semantic relationships, to formulate questions that are not only grammatically correct but also contextually relevant and aligned with the deeper semantic meaning encapsulated in the text[11]. The integration of sentence parsing plays a pivotal role in semantic-based AQG. Advanced parsing methods contribute to the extraction of semantic structures, entities, and relationships within sentences. This, in turn, enhances the precision and context-awareness of the questions generated, moving beyond surface-level understanding to capture the nuanced semantics of the text. Analyzing sentences, especially when improved for semantic comprehension, greatly enhances the accuracy of generating contextual questions[12]. This is achieved by capturing the entities and relationships that are contextually relevant within the sentences.

### 1.0.3 Challenges of Automatic Question Generation

AQG is a challenging task that involves creating natural and contextually relevant questions from given content. Annotated datasets for training AQG models are often limited. This scarcity makes it challenging to build models that generalize well across different domains and contexts. Addressing these challenges involves a combination of advanced natural language processing techniques, machine learning models, and domain-specific knowledge. Ongoing research in these areas aims to enhance the capabilities of AQG systems. The process of automatically generating factual questions for reading assessment involves several computational and linguistic challenges.

### 1.0.4 Shortcomings of Current Sentence Parsing

Online learning is becoming more common and it allows students to access online materials anywhere at any time. In this information era, many organizations and institutions provide a variety of training alternatives to their employees or learners. Due to the radical expansion of the internet over the past two decades, more individuals have got access to online resources [13]. As a result of this development, e-learning is quickly gain popularity as a teaching method, particularly in higher education. The assessment phase, which is used to gauge the academic success of the students, is one of the key difficulties in e-learning. The process of automatically creating questions from different inputs, such as raw text, databases, semantic representations, ontologies, taxonomies, knowledge bases, images, or audio and videos known as AQG.

Researchers [14], from various fields have recently demonstrated a common interest in using AQG for educational reasons. An important function in educational assessment played by AQG is to generate questions and their answers. According to the survey [15], the main challenges in AQG development are the following issues: question generation from multiple sentences, short and long-type answer assessment, question generation, and assessment using machine learning tools. In this study, I focus on the presentation of a neural network-based approach. Complex Sentence Structures add another layer of difficulty, especially in sentences with intricate constructions. Parsing tools may find it challenging to discern specific roles in complex constructions where adverbs modify different elements. The issue of Newly Coined Words and Expressions arises due to the dynamic nature of language, where parsing tools may not be equipped to handle newly coined adverbs or those used in novel ways. Lack of Pragmatic Understanding is also a concern, as parsing tools may struggle to infer implied meanings or the speaker's intent, impacting their ability to identify adverb subtypes correctly.

# Chapter 2

# Thesis of the Dissertation

## 2.1 Thesis 1.

*A novel extended dependency parsing technique has been developed. For the proposed system, I have developed two algorithms that address the current limitations of sentence parsing, which depict the essential steps in our methodology: Algorithm 1: Ruleset Mapping for Question Generation, which selects the best matching rule. Algorithm 2: Question Word Selection for Question Generation to determine the appropriate question word (Wh QTypeWord) based on inputs, including NER, adverb subtype, noun subtype, and dependency tags. This extended dependency parsing method emerges as a promising avenue for enhancing the accuracy and effectiveness of sentence parsing in text-to-semantic applications. The test results show that the proposed algorithm provides questions with acceptable quality.* **[2][5][10][12][13]**

### 2.1.1 Challenges of Dependency Parsing for AQG

As the field of AQG continues to evolve rapidly, future research should focus on developing more advanced models that can generate a wider range of questions, especially for complex sentence structures. The current system is a valuable foundation for further advancements in AQG, offering potential applications beyond educational settings. The implications of this research extend beyond the immediate scope, providing a stepping stone for future AQG developments. The following sections will demonstrate how integrating dependency parsing, NER, adverb, and noun subtype analysis improves the identification of target concepts and question words. Furthermore, I discuss how these improvements impact the quality of the generated questions.

This study proposes an innovative Extended Dependency Parsing approach. This method extends traditional dependency-parsing techniques by incorporating additional contextual and semantic information. By enriching the parsing process, the aim is to improve the accuracy of dependency-based structures, consequently enhancing the quality of questions generated through AQG. One notable challenge of AQG is the complexity of generating meaningful questions from parsed dependencies. Achieving this requires addressing issues such as ambiguous syntactic structures and ensuring the coherence of the generated questions.

### 2.1.2 Sentence Parsing with Extended Dependency Parsing

The proposed system presented a rule-based AQG system that utilized dependency parsing and a comprehensive analysis of English sentence structure. The system has been evaluated using both automatic and human evaluation techniques, and the results showed that the quality of the generated questions was highly dependent on the complexity of the sentence, with better quality and more natural questions generated for sentences with simple structures. Recent advances in AQG have led to the introduction of new models that utilize machine learning techniques, including neural networks, to generate questions from the text. These models can generate questions from both single sentences and paragraphs and have the potential to generate more complex and diverse questions. Furthermore, machine learning techniques, including neural networks, have been applied to question-generation models for various domains, including medical and scientific question generation. In conclusion, the field of AQG is rapidly evolving, and future work will likely focus on developing more advanced models that can generate more diverse and complex questions. The current rule-based system presented in this paper serves as a baseline for future research in the field. The new scientific findings of this chapter are summarized as follows

### 2.1.3 Method

Our methodology is grounded in a strategic fusion of dependency tree parsing and NER techniques. These choices are underpinned by their proven effectiveness and versatility in addressing the core challenges outlined in the introduction. Here, I have provided the necessary details, algorithms, and techniques to allow readers to confirm and replicate our findings. In this regard, dependency tree parsing is a cornerstone of our approach and provides the means to analyze the grammatical structure of sentences by establishing dependency relations between words. The choice of dependency parsing is justified by its inherent ability to handle various language constructs and ambiguous inputs effectively. NER is another integral component of our methodology. NER automates the extraction of valuable information from unstructured natural language documents by categorizing named entities into predefined groups.

For this study, spaCy NER was employed as a fast, statistical, and open-source named entity visualizer. The system assigns labels to groups of contiguous tokens, which encompass named or numerical entities, including person, organization, language, and event, among others. Our proposed system is illustrated in Figure 2.1 and is categorized into distinct modules: Pre-processing: The initial module, involving the removal of stop words and tokenization of the remaining words from the input sentence. NER, POS, and Dependency Parsing: The subsequent modules process the tokenized data, identifying named entities, extracting POS tags, and performing dependency parsing. These elements form the foundation for subsequent stages. The output of this module serves as input for the NER, POS, and dependency parsing modules. The NER module identifies named entities within the input, while the POS module extracts the noun components of the sentence, which are also essential for the ruleset mapping and question generation stages.

The Ruleset adopted from previous work [16] is extended to include named entities, POS tags, and dependency parsing. This enhancement acknowledges the importance of these elements for generating high-quality questions. However, the main limitation of the ruleset was its lack of categorization for adverbs and noun types. To address this limitation, I have developed Algorithm 1 and Algorithm 2, which depict the essential steps in our methodology: Algorithm 1: Ruleset Mapping for Question Generation: This algorithm maps rules to dependency tag lists and selects the best matching rule. It is an essential component of our innovative approach. Algorithm 2: Question Word Selection for Question Generation: This algorithm determines the appropriate question word (Wh_QTypeWord) based on inputs, including NER, adverb subtype, noun subtype, and dependency tags. This step contributes significantly to question generation.

**Algorithm 1** Ruleset mapping for question generation algorithm

```
function RulesetMapping(Ruleset, List_of_Sent_DependencyTag)
    Input: Ruleset, List_of_Sent_DependencyTag
    Output: Question, Answer
    QuestionList ← empty
    for R ← Rule to Ruleset do
        Sim ← similarity(Rule, DependencyTagList)
        if Sim is in BestSimilarityScore then
            WinnerRule ← Rule
            BestSimilarityScore ← Sim
        end if
    end for
    QuestionList ← apply(WhQTypeWord, WinnerRule, DependencyTagList)
    Return QuestionList
end function
```

In response to the limitations of conventional rule-based systems, our methodology innovatively integrates word2vec a powerful word embedding technique. This integration augments the flexibility and effectiveness of our system, making it applicable across diverse domains. The authors noted that the state-of-the-art best match analysis calculation is commonly used to perform rule-set matching.

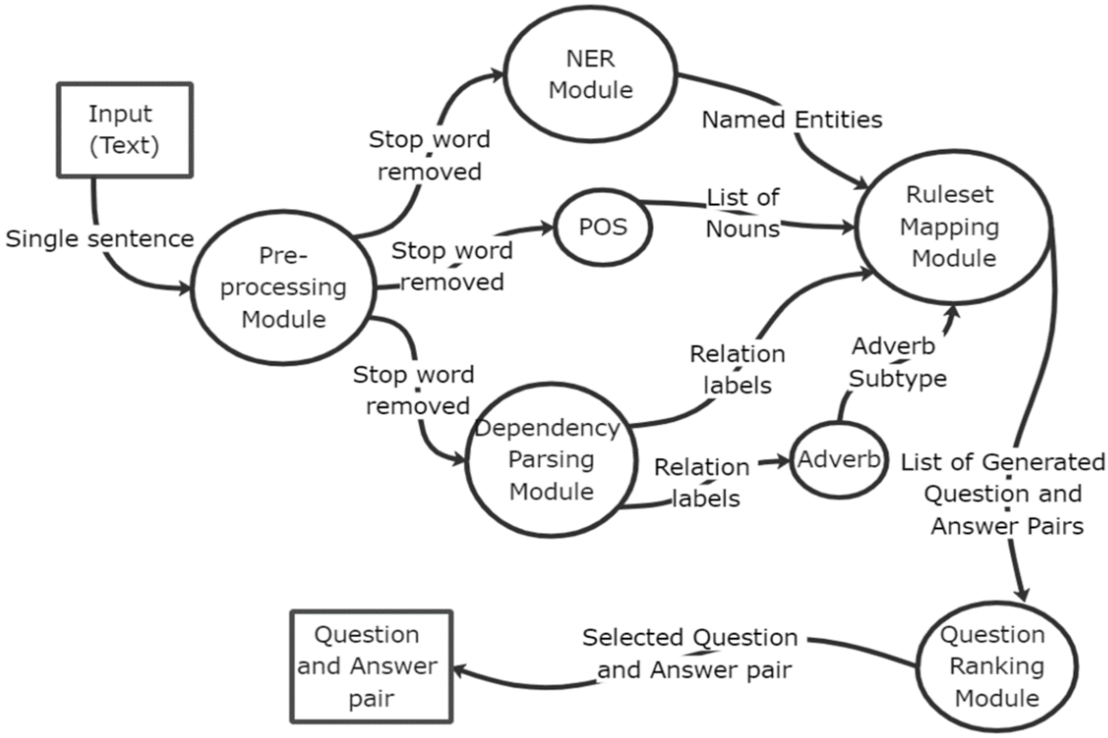


Figure 2.1: Proposed system block diagram

Nevertheless, this mechanism for selecting the best match is rigid, and there are numerous scenarios in which sentences may express the same meaning but are written differently.

A distinctive feature of our approach is the inclusion of adverb subtypes (Time, Place, Manner, Degree, and Frequency) and noun subtypes (Human, Animal, and Thing) for question generation. These subtypes play a pivotal role in crafting high-quality questions. I have provided comprehensive tables (Table 2.1 and Table 2.2) that detail the combinations of these subtypes with their corresponding question words.

## 2.2 Thesis 2.

*A novel MLP-based Sentence Parsing Model has been developed and used for the improvement of parsing accuracy. The model can handle complex linguistic structures and it is in general more effective in generating questions than the rule-based approaches. The developed MLP-based approach emerges as a promising avenue for enhancing the capabilities of AQG.* **[3][4][13][14][15]**

Exploring the architecture of the MLP model designed for sentence parsing in AQG. Understanding the layers, nodes, and activation functions that contribute to the model's learning and inference processes. In this study, I have defined metrics

**Algorithm 2** Algorithm for Question word Selection for Automatic Question Generation

```
function QuestionGeneration(Ruleset, List_of_sent_NER, List_of_sent_AdverbSubType, List_of_sent_NounSubType, List_of_sent_DependencyTag)
    Input: Ruleset, List_of_sent_NER, List_of_sent_AdverbSubType, List_of_sent_NounSubType, List_of_sent_DependencyTag
    Output: Wh_QType
    Set DependencyTagList ← List_of_sent_DependencyTags
    Set QuestionList ← Empty
    BestSimilarityScore ← empty
    Wh_QTypeWord ← empty
    BestScore ← empty
    if List_of_sent_NER is not empty then
        if List_of_sent_NER == "PERSON" then
            Wh_QTypeWord ← "Who"
        else if List_of_sent_NER == "LOC" then
            Wh_QTypeWord ← "What"
        else if List_of_sent_NER == "DATE" then
            Wh_QTypeWord ← "When"      ▷ ... add more conditions based on NER types
        end if
    else if List_of_sent_AdverbSubType is not empty then
        if List_of_sent_AdverbSubType == "PLACE" then
            Wh_QTypeWord ← "Where"
        else if List_of_sent_AdverbSubType == "TIME" then
            Wh_QTypeWord ← "When"
        else if List_of_sent_AdverbSubType == "MANNER" then
            Wh_QTypeWord ← "How"
        else if List_of_sent_AdverbSubType == "FREQUENCY" then
            Wh_QTypeWord ← "How Often"      ▷ ... add more conditions based on AdverbSubType
        else      ▷ Handle other cases
        end if
    else
        if List_of_sent_NounSubType == "PERSON" then
            Wh_QTypeWord ← "Who"
        else if List_of_sent_NounSubType == "ANIMAL" then
            Wh_QTypeWord ← "What"
        else if List_of_sent_NounSubType == "OBJECT" then
            Wh_QTypeWord ← "Which"      ▷ ... add more conditions based on NounSubType
        else      ▷ Handle other cases
        end if
    end if
    Return Wh_QTypeWord
    RulesetMapping(/* Arguments for RulesetMapping function */)
end function
```

Table 2.1: Noun SubTypes and Corresponding Question Words

| Noun SubType | Question Word |
|---|---|
| Human | Who |
| Animal | What |
| Thing | What |

Table 2.2: Adverb SubTypes and Corresponding Question Words

| Adverb SubType | Question Word |
|---|---|
| Time | When |
| Place | Where |
| Manner | How |
| Degree | How |
| Frequency | How often |

to assess the performance of the MLP-based model compared to template-based approaches. Metrics include accuracy, precision, recall, and F1 score in the context of sentence parsing for AQG. I have developed our proposed system using Google Colaboratory, or "Colab" for short, which allows us to write and execute Python in our browser with no additional configuration. Then I divided the dataset into a training set (90 percent) and a testing set (10 percent) using random sampling techniques. The implementation of the template-based and MLP-based question generation is available on GitHub[1].

After I have implemented the proposed system, I need to measure and compare its efficiency. According to our observation, most scholars do not know which methodologies to use for the evaluation techniques of question generation. It is hard to quantify the generated question as "good" because good questions tend to be significant, syntactically correct, semantically sound, and natural. As a result, recent QG research tends to utilize human evaluation. However, human evaluation can be labor-intensive, time-consuming, inconsistent, and hard to reproduce. Due to these, researchers[17] still use automatic evaluation metrics, even though studies have shown that automatic evaluation metrics do not correlate well with fluency and coherence.

In our evaluation methodology, I have used human raters to blindly compare automatically-generated questions with human-generated (golden questions) rating (1-5) marks for all testing questions and BLEU and ROUGE automatic evaluation metrics. The BLEU is a metric to evaluate a generated sentence to a reference sentence. BLEU was originally created to measure the quality of machine translation with respect to human translation. It computes an N-gram precision difference between the two sequences, as well as a penalty for machine sequences being shorter than human sequences. A perfect match receives a 1.0 score, whereas a perfect mismatch receives a 0.0 value. The most you can do is get a 0.6 or 0.7 on the scale. This score was created primarily to assess the accuracy of automatic

[1] https://github.com/waleligntewabe/MLP-based-AQG

machine translation systems' predictions. On a set of references, BLEU calculates the average n-gram precision. A BLEU-n score is a BLEU score that has been calculated using up to n-grams.

### 2.2.1 Comparison of Template-Based and MLP-Based Approaches

In this experiment, the rule set is constructed for the general domain by considering the most common English question patterns and the different structures of the sentences. Regarding the preprocessing phase the first step is tokenization. Tokenization is the mechanism by which a given expression is split into words or other significant elements called tokens. Another operations steps in the preprocessing phase are sentence segmentation, tokenization, POS tagging, and rule matching. Rule set construction and template matching is based on the POS tag feature vector of the tokens. Rules holds both sentence template and their question template. To apply on concrete sentence, the POS tag feature is determined for matching. Demonstration that the neural network-based method using NLP outperforms the template-based approach in both open world and closed word domains.

Sentence template is given by list of POS tags with position index to differentiate the similar POS tags with in the sentence. e.g. [NN1, VBZ1, VBN1, IN1, NN2]. Question template is given by list of common question words and POS tags with position index to differentiate the similar POS tags with in the sentence e.g. [where, NNS1, VBP1, VBN1, IN1, DT1, NN1].

The next task is to evaluate each generated questions with the original sentences and return the best scorer question as a final result. Finally, the system generates a question with all possible constructed templates. Then the system automatically evaluates each generated question with the given sentence using the BLEU metric and takes the maximum score as the final output question.

**Example 1**

Let us assume the rule set contains the following three rules having different numbers of question templates.

**Rule 1**

$$\text{ST} = ['NNS1',' VBP1',' VBN1',' IN1',' DT1',' NN1'];$$
$$\text{QT} = ['Where' +' VBP1' +' NNS1' +' BN1'+'?']$$

**Rule 2**

$$\text{ST} = ['VBG1',' NN1',' VBZ1',' NNS1'];$$
$$\text{QT} = ['Which' +' VBZ1' +' NNS1'+'?']$$

**Rule 3**

$$\begin{aligned}
\text{ST} &= ['NN1','VBZ1','VBN1','IN1','NN2'];\\
\text{QT1} &= ['How' + \text{NN1} +' VBZ1' +' VBN1'+'?'];\\
\text{QT2} &= [\text{NN1} +' VBZ1' +' VBN1' +' IN1' +' what'+'?'];\\
\text{QT3} &= ['Which' +' VBZ' +' VBN' +' IN' +' NN2'+'?']
\end{aligned}$$

The input sentence is the following:

$$S = \text{Limestone is formed by deposition;}$$

In the first step, it generates the token list and yield the following list

T= 'NN1','VBZ1','VBN1','IN1','NN2'.

Based on the similarity calculation using edit distance, I have got the following similarity values for the rules:

$$\begin{aligned}
\text{sim}(T, \text{ST}_{r1}) &= 0.14,\\
\text{sim}(T, \text{ST}_{r2}) &= 0.08,\\
\text{sim}(T, \text{ST}_{r3}) &= 1,
\end{aligned}$$

Based on the best similarity score, the winner is Rule 3.

Using the substitutions, I have got the concrete variants for the question templates of the winner rule:

$$\begin{aligned}
Q1 &= \text{How limestone is formed?; BLEU scores 0.55}\\
Q2 &= \text{Limestone is formed by what?; BLEU scores 0.668}\\
Q3 &= \text{Which is formed by deposition?; BLEU scores 0.56}
\end{aligned}$$

Based on the BLEU score, the winner question is "Limestone is formed by what?"

**MLP architecture**

MLP is a supplement of a feed-forward neural network and consists of three types of layers the input layer, output layer, and hidden layer. The neural network architecture learns any function $f(\cdot) : \mathbb{R}^m \rightarrow \mathbb{R}^o$ by training on a dataset, where m is the number of dimensions for input and o is the number of dimensions for output.

First, I read and pre-processing the dataset then train the model on the implemented model next train and predict the test data. I have created datasets both manually and from the QASC [2] dataset to prepare the MLP training model. The QASC dataset is a question-and-answer set that focuses on sentence composition. It includes a corpus of 17 million sentences and 9,980 multiple-choice questions regarding grade school science (8,134 train, 926 dev, 920 test). I have discovered several null values in the QASC dataset, as well as very long and useless sentences. Due to this, I have tried to preprocess and clean up the dataset and selected only the top 700 short and more meaningful sentence question pairs. In addition, I have constructed 300 sentence question pairs manually from general truth and common sentences, and finally, built 1000 sentence question pair datasets for our training.

In our experiment, the first preprocessing step is to convert the sentence into a sequence of phrases. Phrases are a combination of two or more words that can take the role of a noun, a verb, or a modifier in a sentence. In the English language, there are five phrase types i.e. NP, VB, ADJP, ADVP, and PP. I have used chunking to extract phrases from sentences. To construct the input matrix for the MLP model build a vocabulary with a combination of English phrases and unique words that exist only in questions. Then I extend the vocabulary with the WH question words and the most frequently unique words. In our test, I built up vocabulary containing 40 unique words. Sentence and question vector representation for MLP Training model.

## 2.3 Thesis 3.

*I have developed two novel sentence parsing approaches that are based on deep neural network technologies. The first method uses the prompted version of ChatGPT while the second applies Hybrid Parser and neural network-based method. Through a comprehensive analysis, the Hybrid Parser-Based approach demonstrates a slight advantage in accuracy compared to ChatGPT in sentence parsing tasks. Notably, the Hybrid Parser consistently maintains "excellent" response quality, showcasing stability across various inputs, while ChatGPT's response quality varies with prompt sizes. The findings contribute to the broader field of natural language processing and offer valuable insights for practical applications, including information retrieval and knowledge graph development.* **[1][7][8][9]**

### 2.3.1 Adverbs in Sentence Parsing

Adverbs play a crucial role in sentence parsing by providing valuable information about manner, time, frequency, and degree [18]. Understanding the role of adverbs in sentence structure and meaning is essential for accurate adverb-type categorization, which has implications for various natural language processing tasks. This section provides an overview of the significance of adverbs in sentence parsing and their impact on language understanding. Adverbs modify verbs, adjectives,

[2] https://github.com/allenai/qasc

and other adverbs in a sentence, influencing the overall semantics and conveying additional details. They provide information about how an action is performed (manner), when an action occurs (time), how often an action happens (frequency), and the intensity or extent of an action (degree). For example, in the sentence "She sings beautifully," the adverb "beautifully" modifies the verb "sings" to indicate how she sings.

### 2.3.2 Hybrid Parser-Based Method

The creation of a Hybrid Parser-based sentence parsing framework is a noteworthy breakthrough in the field of NLP. This innovative approach combines rule-based and machine-learning methods to extract meaning from text, addressing the limitations of current NLP parsing techniques. By incorporating both rule-based and machine-learning components, this framework becomes capable of handling a wider range of linguistic structures and domains, ensuring robust performance. Its primary objective is to enhance the accuracy of semantic parsing by capturing context-specific elements in language, ultimately improving the comprehension of the underlying meaning in the text. The framework strikes a careful balance between accuracy and efficiency, allowing for the precise construction of a semantic graph from textual content. The architecture of this framework encompasses text preprocessing, rule-based and machine learning-based sentence parsing, adverb-type prediction, and semantic graph construction.

One distinguishing feature of this framework is its dedicated component for predicting adverb types within the text. This feature plays a pivotal role in accurately extracting the essence of a sentence. The integration of outputs from both rule-based and machine learning-based parsing yields a comprehensive semantic graph representing the structured knowledge present in the text. This Hybrid parser-based approach harnesses the strengths of rule-based systems, which excel at handling linguistic patterns and prior knowledge, and machine learning models, which adapt to context and data-driven insights. As a result, the framework enhances natural language understanding and information extraction, offering a promising solution to the challenges presented by traditional parsing methods.

### 2.3.3 Adverb Type Categorization

Adverb-type categorization is a crucial task in NLP, playing a significant role in various language understanding tasks. Adverbs provide valuable information about manner, time, frequency, and degree in English sentence parsing, making their accurate categorization essential for understanding sentence structure and meaning. Traditional rule-based approaches and curated dictionaries have been widely used for adverb-type categorization. Its ability to generate contextually relevant and coherent responses makes it a promising candidate for adverb-type categorization. By leveraging its language generation capabilities, ChatGPT 3.5

can potentially capture the contextual information and linguistic patterns necessary for accurate adverb-type predictions.

To evaluate the efficiency of ChatGPT 3.5 for adverb type categorization, a comparative analysis with a dictionary and ML-based method is conducted[19]. This approach combines curated dictionaries of adverb types with ML techniques to classify adverbs into predefined categories. By contrasting the performance of ChatGPT 3.5 with this established method, insights into the strengths and weaknesses of ChatGPT 3.5 can be gained. The evaluation aims to assess the effectiveness of ChatGPT 3.5 in accurately categorizing adverb types and to explore its potential applications in sentence parsing and other language understanding tasks. The findings from this evaluation contribute to a deeper understanding of the capabilities and limitations of ChatGPT 3.5 for adverb-type categorization.

The proposed framework also includes a dictionary that contains some selected words with the related unit type labels:

One of the following parsing processes are Wordnet-based Lin similarity (ll): a score denoting how similar two word senses (s1, s2) are, based on the Information Content (IC) of the Least Common Subsumer (sc) most specific ancestor node) and that of the two input synsets.

$$l_l(s_1, s_2) = \frac{2 \cdot IC(s_c)}{IC(s_1) + IC(s_2)} \tag{2.1}$$

I have divide this dictionary into two parts:

$$D = D_B \cup D_L$$

where $D_B$ is the set of baseline words, I have used to determine the similarity positions of new query words. For a given query word $w_q$, the following local feature vectors are calculated:

$\{l_e(w_q, w), l_l(w_q, w), l_p(w_q, w), l_d(w_q, w), l_w(w_q, w), l_s(w_q, w) | w \epsilon D_B\}$ Using these similarity measures, the generated similarity vectors are merged into a global feature vector

$$l(w_q)$$

These global feature vectors are used to predict the corresponding unit type label of $w_q$. For the prediction, an MLP neural network module (NN) is involved, where outputs the predicted unit label.

$$cat = NN(l(w))$$

### 2.3.4 Results and Analysis

The model initially categorizes words and phrases in a given sentence into different functional structures, such as subject, verb, object, time adverb, place adverb, frequency adverb, and manner adverb. Due to the expensive nature of human-level evaluation, the training dataset was limited to 200 sentences, while the testing dataset consisted of 40 sentences. In this scenario, the dataset is divided into 80% for training and 20% for testing purposes.

Fifteen linguistic teachers from Higher Education institutions evaluate the overall categorization, and they rank the adverb-type categorization results on a scale of 1 (poor) to 5 (excellent). Additionally, they assess the classified adverb type as correct or incorrect. Among the evaluated adverbs, the maximum score is assigned to each adverb. Examples of classifications that were evaluated as partially correct (grades 3 or 4):-

Sentence: The child reads the book carefully and attentively at the library everyday.

- ChatGPT 3.5 Based Method Generated result:- Predicate: read, Subject: child, Object: the book, Time: everyday, Place: at library, Manner: carefully, and attentively.
- Dictionary and ML-based Method Generated result:- Subject: The child, Predicate: reads, Object: the book, Time: everyday, Place: at the library, Manner: carefully and attentively, Frequency: everyday.

The performance of the two experimental models in functional English sentence structure analysis is evaluated using a 1-5 scale, with 5 being the best and 1 being the worst. The evaluation was carried out by 15 linguistic experts who assessed the correct extraction of functional sentence structure by the models. The evaluation metrics helped to determine the effectiveness of the models and their suitability for use in the educational domain. The evaluation sentences were selected based on their functional structure, and a total of 40 sentences were carefully chosen from various sources to ensure representativeness. The evaluation scores range from a minimum of 1.5 to a maximum of 4, showcasing the experts' assessments of the performance of these methods.

Efficiency, as reflected in the average quality rating of responses generated by these models, is a key measure. I have explored prompt set sizes ranging from 5 to 40. The OpenAI API model garnered quality ratings, spanning from "poor" with a prompt set size of 5, to "below average" with 15, "average" with 25, "above average" with 35, and ultimately "excellent" with a prompt set size of 40. Surprisingly, both the ChatGPT 3.5 Web Interface and the Hybrid Parser-based Sentence Parsing model consistently maintained an "excellent" response quality rating, irrespective of the prompt set size. This indicates their enduring efficiency across a spectrum of prompt set sizes. This table provides valuable

Table 2.3: Efficiency of ChatGPT in Dependency of Prompt Size.

| Model | Prompt Size | Average Rating |
|---|---|---|
| OpenAI API | 5 | Poor |
| | 15 | Below Average |
| | 25 | Average |
| | 35 | Above Average |
| | 40 | Excellent |
| ChatGPT 3.5 Web Interface | – | Average |
| Hybrid Parser-based Sentence Parsing | – | Excellent |

insights into how different prompt set sizes impact ChatGPT model efficiency, revealing noteworthy disparities in performance between the OpenAI API and other models. This experiment underscores that while ChatGPT 3.5 is a recent and versatile language model capable of generating diverse and interesting results, it has limitations, particularly in domains like sentence parsing. The observed accuracy values strongly advocate for the effectiveness of the proposed hybrid-parser-based sentence parsing. This suggests that the proposed model may find broader applicability in sentence-parsing tasks.

## 2.4 Thesis 4.

*I demonstrated the efficiency of the proposed method in two application domains. The first domain is the field of automatic question generation and the second refers to the automatic semantic graph induction. The AQG module was developed in Python as a prototype module in the Intelligent Tutoring System. The performance test result shows that the developed framework can be used in real applications.The second module can be used to generate RDF ontology descriptions from the free text data sources using our proposed sentence parser engines. The test result shows that the proposed module can be used to automate the process of ontology generation.* [1][6]

In the area of Sentence Parsing applications, two important examples are AQG in ITS and the creation of Ontology Semantic Graphs. These applications represent practical uses of sentence parsing, frequently employing sophisticated NLP techniques. Let's delve into how these applications are interconnected with the process of sentence parsing: AQG within ITS is a tangible, real-world application of sentence parsing, highlighting the practical significance of advanced NLP techniques. ITS is a computer-based system that aims to offer direct and customized instruction or feedback to learners with personalized guidelines based on their cognitive skills, usually without requiring intervention from a human teacher. Different researchers, designers, and developers define ITSs in different ways. In this application, sentence parsing plays a pivotal role in deconstructing instructional content into its syntactic and semantic components.

```
Subject=John
Verb=ate
Object=an apple
Time Adverb=yesterday
```

Figure 2.2: Functional structure for the the sentence ”John ate an apple yesterday”

This parsing process is instrumental in breaking down sentences into syntactic and semantic elements, laying the foundation for effective question generation. The AQG process begins with the parsing of instructional text, where relevant information is extracted and relationships between different components are understood. This parsed data is then utilized to craft meaningful questions aligned with specific learning objectives. The parsed structure guides question generation, ensuring contextual relevance and contributing to a cohesive learning experience. Thus, sentence parsing serves as a vital beginning to AQG in ITS, showcasing its real-world impact in customizing assessments, delivering prompt feedback, and cultivating a learning environment that is both personalized and adaptive.

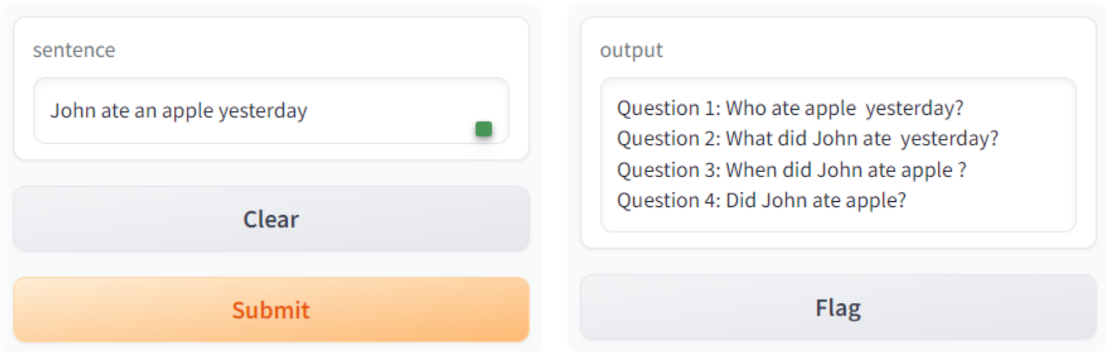


Figure 2.3: Sample Automatic Generated Questions for the sentence ”John ate an apple yesterday”

Figure 2.3 illustrates a set of sample questions that have been automatically generated from the sentence ”John ate an apple yesterday.” The questions showcased in the figure are the result of an automated question-generation process. This process involves analyzing the given sentence and formulating relevant questions to assess comprehension or generate educational content. I have used Gradio for building GUI web applications. In the context of AQG, sentence parsing proves critical for comprehending grammatical structures and extracting meaningful components, such as subjects, predicates, and objects.

### 2.4.1 Basics of Semantic Graph

A semantic graph is a graph model where nodes represent concepts and edges (or arcs) represent relationships between those concepts. This model type is often used in artificial intelligence applications for representing knowledge.

**Definition 2.1**

A graph $G = (V, E)$ is defined by a set of nodes $V$ and a set of edges $E$ between these nodes, and a set $E \subseteq V \times V$ of directed edges (or arcs). An edge going from node $u \in V$ to node $v \in V$ is denoted as $(u, v) \in E$ and has a start (tail) vertex $u$ and an end (head) vertex $v$.

Building semantic graphs is essential for many practical uses and ongoing research . As I have more and more data available, creating these meaningful graphs becomes increasingly important for learning from different sources. Scientists keep looking for new ways to make this field better, and they use it in things like understanding language, organizing knowledge, and using artificial intelligence. They make structured graphs and networks to show how words, ideas, and things are connected. These graphs help in finding information, answering questions, and suggesting things you might like. So, making these graphs is a big part of helping computers and people work together better. When texts are represented graphically, it allows the preservation of additional information like the text's inner structures, semantic relationships, and term order. However, events like these are not effectively captured using current NLP parsing and semantic graph construction. Researchers are actively exploring the creation of these graphs and how they can represent knowledge, diving into structured data, relationships, and more detailed elements, which align with prior work on SRL and adverb sense disambiguation. These efforts aim to provide a more comprehensive understanding of semantic parsing, event descriptions, and the complexities involved, as outlined in related works.

### 2.4.2 Generation of Ontology Semantic Graphs

Regarding the implementation of an ontology and NLP engine, Python is one of the most common languages used. It is an interpreted, object-oriented, extensible programming language, which provides an excellent combination of clarity and versatility in different disciplines. Information science offers many modules and packages for management and implementing ontology, data mining, and NLP. Many tools are available for building or managing an ontology. Regarding editing of the ontology by humans, Protégé is the most commonly used editor framework, which was created at Stanford University [20]. Protégé is free, open-source software to construct and update the ontology knowledge base. The tool has features for building, editing, and visualizing ontologies and importing and exporting capabilities of different ontology formats.

In Figure 2.4, we can see a sample RDF graph corresponding to the sentence "Lalibela stands as an ancient rock-hewn church in Ethiopia." The RDF graph visually represents the structured information derived from the sentence using the RDF format. Each node and link in the graph signifies a distinct element or relationship present in the sentence. This graphical representation offers a clear illustration of how the sentence's content is encoded into RDF, facilitating a more organized and standardized representation of information.

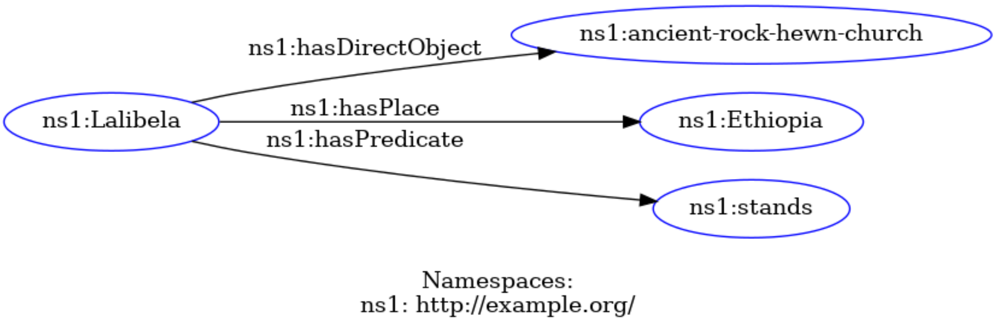


Figure 2.4: Sample RDF graph for sentence "Lalibela stands as an ancient rock-hewn church in Ethiopia"

# Author's Publications

## 1 Journal Articles Related to the Dissertation


1. Sewunetie, W. T. & Kovács, L. Exploring Sentence Parsing: OpenAI API-Based and Hybrid Parser-Based Approaches. *IEEE Access* **12,** 38801–38815. doi:10.1109/ACCESS.2024.3360480 (2024).
2. Tewabe, W. & Kovacs, L. A Comparative Study of ChatGPT-based and Hybrid Parser-based Sentence Parsing Methods for Semantic Graph-based Induction. *International Journal of Advanced Computer Science and Applications (IJACSA)* **15.** doi:10.14569/IJACSA.2024.01501117. http://dx.doi.org/10.14569/IJACSA.2024.01501117 (2024).
3. Sewunetie, W. T. & Kovács, L. Automatic Question Generation Based on Sentence Structure Analysis and Dependency parsing approach. *Indonesian Journal of Electrical Engineering and Computer Science* **32.** Scopus Q3 Accepted for publication. doi:http://doi.org/10.11591/ijeecs.v32.i3 (2023).
4. Sewunetie, W. T. & Kovács, L. Comparison of template-based and multilayer perceptron-based approach for automatic question generation system. *Indonesian Journal of Electrical Engineering and Computer Science* **28.** Q3 Indexed, 1738–1748. doi:http://doi.org/10.11591/ijeecs.v28.i3.pp1738-1748 (2022).
5. Sewunetie, W. T. & Kovács, L. Evaluation of Python-Based NLP Frameworks for Template-Based Automatic Question Generation. *Production Systems and Information Engineering* **9,** 54–66. ISSN: 1785-1270. https://psaie.iit.uni-miskolc.hu/PSAIE2020.pdf (2020).
6. László, K. & Tewabe, S. W. Challenges in target selection for automatic question generation. *Multidiszciplináris Tudományok* **9,** 500–507. doi:https://doi.org/10.35925/j.multi.2019.4.53 (2019).
7. Tewabe Sewunetie, W., Hussein Ali Ahmed, G. & Kovács, L. The development and analysis of extended architecture model for intelligent tutoring systems. *GRADUS* **6,** 128–138. http://gradus.kefo.hu/archive/2019-4/2019_4_CSC_014_Sewunetie.pdf (2019).

# 2 Conference Proceedings Related to the Dissertation


8. Sewunetie, W. & Kovács, L. *ChatGPT in Education: Opportunities and Concerns for Functional English Sentence Structure Analysis for Automatic Question Generation* in *Proceedings of the International Conference on Advanced Science and Technology (ICAST)* Scopus Indexed Accepted Paper (2023). https://icast-conf.eai-conferences.org/2023/accepted-papers/.
9. Kovács, L., Csépányi-Fürjes, L. & Tewabe, W. *Transformer Models in Natural Language Processing* in *Proceedings of the INTER-ENG 2023 Conference* Scopus Accepted Paper (Springer, 2023). https://inter-eng.umfst.ro/2023/.
10. Sewunetie, W. T. & Kovács, L. *Efficiency Evaluation of ChatGPT for Adverb Type Categorization* in *2023 World Symposium on Digital Intelligence for Systems and Machines (DISA)* (2023), 199–203. doi:10.1109/DISA59116.2023.10308928.
11. Sewunetie, W. T. & Kovács, L. *Automatic Question Generation based on Sentence Structure Analysis* in *2023 24th International Carpathian Control Conference (ICCC)* Scopus Indexed (2023), 411–414. doi:10.1109/ICCC57093.2023.10178946. https://ieeexplore.ieee.org/document/10178946.
12. Sewunetie, W. T. & Kovács, L. *Comparison of Automatic Question Generation Techniques* in *2022 IEEE 22nd International Symposium on Computational Intelligence and Informatics and 8th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics (CINTI-MACRo)* Scopus Indexed (2022), 000025–000030. doi:10.1109/CINTI-MACRo57952.2022.10029559.
13. Sewunetie, W. T. & Kovacs, L. *Rule-Based Automatic Question Generation Using Dependency Parsing* in *The 13th Conference of PhD Students in Computer Science* (2022), 192–196. https://www.inf.u-szeged.hu/~cscs/pdf/cscs2022.pdf.
14. Sewunetie, W. T. & László, K. *MLP Based Model for POS-Level Automatic Question Generation using Keyword Extraction* in *Abstract Book for the 17th Miklós Iványi International PhD & DLA Symposium: Architectural, Engineering and Information Sciences* (Pollack Press, Pécs, 2021).
15. Kovacs, L. & Walelign, T. S. *Automatic Question Generation using Template-Based Shallow NLP Approach* in *Doktorand Forum* (Nov. 2020).
16. László, K. & Sewunetie, W. *Closed Domain Template-Based Question Generation* in *16th Miklós Iványi International PhD & DLA Symposium* (2020), 127.
17. Kovács, L. & Walelign, T. S. *Evaluation of Template-Based Automatic Question Generation Techniques* in *XXIII. Tavaszi Szél Konferencia Absztrakt Kötet : Association of Hungarian PHD and DLA Students* (2020), 364. https://www.dosz.hu/__doc/dokumentumfile/2020/tavaszi-szel-2020-absztrakt-kotet-i-final.pdf.

18. Kovács, L. & Walelign, T. S. *A Comparative Survey of Automatic Question Generation Techniques* in *Doktoranduszok Fóruma* (Nov. 2019), 173–176.

# 3 Other Journal Articles

1. Gebretsadik, K. K. & Sewunetie, W. T. *Designing Machine Learning Method for Software Project Effort Prediction* 2019. doi:10.5923/j.computer.20190901.02.
2. Walelign, T. & Eshete, D. *Design and Architecture of Desktop LCD Monitor Having Built-in Webcam and Microphone with Multi-Angled Flexibility* 2016.
3. Walelign, T. & Eshete, D. *Design and Development of a Real-Time Virtualized Cloud-Based Mobile Phones Data Storage for Ethio Telecom* 2016. doi:DOI:10.9790/0050-03044751. https://api.semanticscholar.org/CorpusID:55505075.
4. Emiru, E. D. & Sewunetie, W. T. *Automatic Speech Segmentation for Amharic Phonemes Using Hidden Markov Model Toolkit (HTK)* 2016.
5. Sewunetie, W. T. & Emiru, E. D. *Real-Time Speech Synthesis by Handling Amharic Gemination Words Using Statistical Technique* 2016. https://api.semanticscholar.org/CorpusID:212510211.

# 4 Other Conferences

6. Sewunetie, W. T. *Amharic Text-to-Speech System* 2023. https://openreview.net/forum?id=h2GFfN9Oec.
7. Sewunetie, W. T. *et al.* *Gender Bias Evaluation in Machine Translation for Amharic, Tigrigna, and Afaan Oromoo* Accepted. Sheffield, UK, June 2024.
8. Sewunetie, W. *et al.* *Large Language Models for Sexual, Reproductive, and Maternal Health Rights* Accepted. Paper Status: In production. Orlando, June 2024.
9. Sewunetie, W. T. *et al.* *Evaluating Gender Bias in Machine Translation for Low-Resource Languages* Accepted, Co-located with ICLR 2024, May 11th, 2024, Messe Wien Exhibition Congress Center, Vienna, Austria. 2024.
10. Azime, I. A. *et al.* *Enhancing Amharic-LLAMA: Integrating Task Specific and Generative Datasets* Accepted, Co-located with ICLR 2024, May 11th 2024, Messe Wien Exhibition Congress Center, Vienna, Austria. 2024.
11. Sewunetie, W. *et al.* *Development of A Virtual Store over Cloud for Mobile Phone and SIM Card Data* Raj Kumar Goal Institute of Technology, 2013.

# References


1. Danon, G. & Last, M. A syntactic approach to domain-specific automatic question generation. *arXiv preprint arXiv:1712.09827* (2017).
2. Fasya, E. L. *Automatic question generation for virtual humans* MA thesis (University of Twente, 2017).
3. Alsubait, T., Parsia, B. & Sattler, U. Ontology-based multiple choice question generation. *KI-Künstliche Intelligenz* **30,** 183–188 (2016).
4. Cohen, F. S. What is a Question? *The monist,* 350–364 (1929).
5. Heilman, M. & Smith, N. A. *Good question! statistical ranking for question generation* in *Human language technologies: The 2010 annual conference of the North American Chapter of the Association for Computational Linguistics* (2010), 609–617.
6. Mostow, J. & Chen, W. *Generating Instruction Automatically for the Reading Strategy of Self-Questioning.* in *AIED* (2009), 465–472.
7. Serban, I., Sordoni, A., Bengio, Y., Courville, A. & Pineau, J. *Building end-to-end dialogue systems using generative hierarchical neural network models* in *Proceedings of the AAAI conference on artificial intelligence* **30** (2016).
8. Zhou, Q. *et al. Neural question generation from text: A preliminary study* in *Natural Language Processing and Chinese Computing: 6th CCF International Conference, NLPCC 2017, Dalian, China, November 8–12, 2017, Proceedings 6* (2018), 662–671.
9. Das, B., Majumder, M., Phadikar, S. & Sekh, A. A. Automatic question generation and answer assessment: a survey. *Research and Practice in Technology Enhanced Learning* **16,** 1–15 (2021).
10. Yousuf, H., Lahzi, M., Salloum, S. A. & Shaalan, K. A systematic review on sequence-to-sequence learning with neural network and its models. *International Journal of Electrical & Computer Engineering (2088-8708)* **11** (2021).
11. Fattoh, I. E. Semantic based automatic question generation using artificial immune system. *Computer Engineering and Intelligent Systems* **5,** 74–82 (2014).
12. Deng, Y. *et al.* Contextualized knowledge-aware attentive neural network: Enhancing answer selection with knowledge. *ACM Transactions on Information Systems (TOIS)* **40,** 1–33 (2021).

13. Säljö, R. Digital tools and challenges to institutional traditions of learning: technologies, social memory and the performative nature of learning. *Journal of computer assisted learning* **26,** 53–64 (2010).

14. Kurdi, G., Leo, J., Parsia, B., Sattler, U. & Al-Emari, S. A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education* **30,** 121–204 (2020).

15. Das, B., Majumder, M., Phadikar, S. & Sekh, A. A. Automatic question generation and answer assessment: a survey. *Research and Practice in Technology Enhanced Learning* **16(1),** 1–15 (2021).

16. Sewunetie, W. T. & Kovács, L. *Comparison of automatic question generation techniques* in *2022 IEEE 22nd international symposium on computational intelligence and informatics and 8th IEEE international conference on recent achievements in mechatronics, automation, computer science and robotics (CINTI-MACRo)* (2022), 000025–000030.

17. Nema, P. & Khapra, M. M. Towards a better metric for evaluating question generation systems. *arXiv preprint arXiv:1808.10192* (2018).

18. Chauhan, S. Comprehensive review of coronavirus disease 2019 (COVID-19). *Biomedical journal* **43,** 334–340 (2020).

19. Gondaliyaa, C. P., Patel, A. M. & Parikh, S. M. *A Comparative Study on Machine Learning Based Algorithms* in *Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT)* (2018), 26–27.

20. Horridge, M. *et al. Simplified OWL ontology editing for the web: is WebProtégé enough?* in *The Semantic Web–ISWC 2013: 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I 12* (2013), 200–215.