Bachelor of Arts in Linguistics thesis

Syntactic Complexity and LLM Performance: Exploring the Role of Prompt Design

December 2024

Supervisor: Sangah Lee

Department of Linguistics

College of Humanities

Seoul National University

Saehee Eom

Abstract

Large Language Models (LLMs) have recently attracted attention in the field of Natural Language Processing (NLP) and have demonstrated outstanding performance in various tasks. This paper focuses on the ability of LLMs to handle syntactic complexity and explores the impact of prompt design on this performance. To this end, we evaluate the model's comprehension and generation capabilities using a set of sentences with varying syntactic complexity and analyze how appropriate prompt design can improve model performance.

Our research results revealed that syntactic complexity is a key factor influencing model performance, and that part-of-speech diversity significantly impacts the model's comprehension and predictive capabilities. Various prompt design techniques, particularly instruction tuning and three-shot learning, were useful in significantly improving performance in specific tasks. This study highlights the importance of prompt design for optimizing LLM performance and suggests that future research should focus on constructing datasets for more precise correlation analysis .

Table of Contents

1 Introduction

Artificial Intelligence (AI) is a model designed to mimic human intelligence. Just as humans struggle to understand long sentences or complex grammatical structures, AI models also exhibit limitations in processing such sentences. This raises the need for research to identify the limitations of AI models' language processing capabilities and to improve them.

Syntactic complexity is a key indicator of language processing capabilities, indicating the degree to which sentences are composed of complex elements and diverse structures . This study limits the scope of discussion to large language models (LLMs) among AI models and evaluates the correlation between the language processing capabilities of these models and the syntactic complexity of input text. Furthermore, we analyze the role of appropriate prompt engineering in overcoming the limitations of these cutting-edge models. This study aims to answer the following questions:

a. How does LLM handle sentences with high syntactic complexity?
b. prompt design affect the handling of syntactic complexity in LLM?

2 Theoretical Background

2.1 Concept and Key Features of Large Language Models (LLMs)

LLM stands for "Large Language Model," an AI model designed to operate in a manner similar to human language understanding by learning from massive amounts of text data. The term "Large" comes from the sheer scale of the parameters these models use, which can range from billions to trillions. Beyond simply listing words, LLMs can analyze the structure and meaning of sentences and perform complex linguistic tasks based on this analysis .

These technologies have evolved gradually, starting from early rule-based natural language processing (NLP) approaches. The emergence of Transformer-based Bidirectional Encoder Representations from Transformers (BERT) in 2018 marked a paradigm shift in NLP, leading to the subsequent development of more powerful LLMs such as GPT-2 and GPT-3 (Brown et al., 2020). Since then, more recent models such as OpenAI's GPT-4 and Google's PaLM have demonstrated outstanding performance in areas such as multilingual processing, advanced contextual understanding, and creative text generation. These models have produced groundbreaking results in diverse tasks such as machine translation, document summarization, question answering, and coding assistance, establishing themselves as core technologies in modern NLP.

LLM has established itself as one of the most promising tools throughout the history of NLP and is also a key research topic in linguistics and cognitive science. While early NLP techniques relied on rule-based approaches, LLM focuses on learning from massive amounts of data, identifying statistical patterns, and generalizing them. This advancement is deeply connected to the linguistics goal of emulating and understanding the complexity of human language, broadening the possibilities for the interaction between language model research and linguistics.

2.2 Definition and measurement indicators of syntactic complexity

Syntactic complexity is a multidimensional concept that encompasses sentence length, the number of subordinate clauses, and grammatical structure (Lu, 2010). For example, sentence length can be assessed by word count or the complexity of its constituent elements. When expressed in a tree structure, the tree depth can serve as an indicator of complexity. The more subordinate clauses there are, the greater the tree height and branches, making analysis

more difficult. For example, a sentence like "The book that the teacher who you met yesterday recommended is fascinating" contains numerous subordinate clauses, demonstrating high syntactic complexity. These factors are utilized to assess the language development of second language learners and analyze the readability of text (Corpus-Based Evaluation of Syntactic Complexity Measures, 2010). Furthermore, they can serve as important criteria for evaluating the processing capabilities of large-scale language models.

2.3 Prompt Engineering

Prompt engineering is the process of designing the text provided as input to an LLM. Prompts are a crucial element that guide the model to perform a given task, and the model's performance can vary significantly depending on the format and content of the input. Prompts can take various forms, such as questions, commands, and explanations, and the specificity and context provided by the prompts directly impact the quality of the model's output.

There are three main approaches to prompt engineering: ▲ Using directives (providing clear and specific instructions to the model to achieve the desired result), ▲ Providing context (including background information about the question), and ▲ Using examples (In-Context Learning: providing examples of tasks to guide the model to understand and perform the task). Among them, the last approach is the one chosen by many researchers. Brown et al. (2020) also demonstrated the effectiveness of this approach through experiments targeting the GPT - 3 model.

3 Experimental setup

3.1 Task

this study, three key tasks were selected to evaluate the performance of LLMs. The goal was to comprehensively evaluate model performance across various aspects.

First, the Yes/No question answering task focuses on evaluating the model's simple binary classification ability. This task measures the model's ability to answer a given question with "yes" or " no ." For example, given the passage "The cancellation of The Border was announced by the CBC after three seasons were aired," and the question "is there going to be a season 4 of the border?", the model can accurately respond. This task is useful for assessing the model's ability to understand context and generate an appropriate binary response. This allows us to verify the model's basic reasoning ability and understanding.

Second, the text completion task assesses the model's contextual understanding and generation capabilities. It measures how the model completes the remaining sentences based on a given part of the sentence. For example, given the sentence "A fan created a petition...", it assesses whether the model can generate an appropriate completion that fits the context, such as "at the official White House petition site." This task is a crucial way to assess the model's ability to generate natural and coherent text based on previous context.

Third, the English-Korean translation task evaluates the model's cross-language translation capabilities. This task measures the model's ability to accurately translate English sentences into Korean. For example, it evaluates the model's ability to translate the sentence "Thank you so much for coming" into "와줘서 정말 고마워요." This task analyzes how well the model can maintain syntactic and semantic correspondence between the two languages, with a particular focus on grammatical consistency and natural expression. This allows for the evaluation of the model's multilingual processing capabilities and translation performance.

3.2. Model Selection

Computational resource limitations, this study selected relatively small models as candidates, ultimately selecting those that demonstrated a certain level of performance across all tasks. The research environment used was a Google Colaboratory T4 GPU (15GB VRAM). Because experiments had to be conducted within limited computing resources, efficient model selection was essential. The candidate models were selected as follows:

meta-llama/Llama-3.2-1B-Instruct
mistralai /Mistral-7B-Instruct-v0.3
Qwen/Qwen2.5-1.5B-Instruct
google/gemma-2-2b-it

These models are open source on the Huggingface platform and are free, making them more economical than API-based models. Using each model, we attempted to perform all three tasks: yes/no question answering, text completion, and English-Korean translation. Ultimately, we selected a model that performed well on all three tasks when fed general text data. I chose gemma-2-2b- it.

3.3 Dataset and Evaluation Criteria

The datasets and evaluation criteria used in the experiment are as follows.

A, Yes/No question response

BoolQ We used the BoolQ dataset (validation set, 101 rows) and the GardenPath QA dataset (84 rows). BoolQ is a Yes/No question answering dataset created by Google, consisting of naturally generated questions and related documents. Each example consists of a set of questions, text, and answers . The GardenPath QA dataset is It consists of Garden Path Sentences. Garden Path Sentences are sentences whose first interpretation when reading the sentence and the subsequent interpretation after reading the entire sentence are different due to their sentence structure, such as “The old man the boats.” This dataset was constructed using 42 sentence pairs presented by Sturt et al. and Grodner et al.

B. Text completion

The Salesforce/Wikitext (test set, 6,656 rows) dataset was used for text completion task. The Wikitext dataset is a text completion dataset generated from Wikipedia articles and was used to evaluate the model's ability to complete the remainder of a given sentence.

C. English-Korean translation

For the English-Korean translation, we used the MSARMI9/Korean-English Multi-target TED Talks (test set, 1,979 rows) data set. This dataset consists of English-Korean translation data collected from TED Talks. It covers a wide range of topics, making the results of our translation performance measurements highly generalizable.

All datasets were used as test/validation sets to prevent data leakage, which occurs when the model infers data that has already been seen during the model's training process, and to enable fair evaluation. Appropriate metrics were used for each task as evaluation measures. For the Yes/No question answering task, accuracy, which is the proportion of examples correctly predicted by the model, was used. For the text completion task, perplexity was used. For the English-Korean translation task, BLEU score, which measures the similarity between the translation result and the reference translation, was used.

3.4 Prompt Design

The prompt design primarily used the developer-recommended prompts and chat

templates. The model was trained using the following prompt templates :

```
messages = [
    { " role": "user", "content": "You are a helpful assistant. Please create the answer to the questions directly without any explanation. Do you understand? " },
    { " role": "assistant", "content": "Yes." },
    { "role": "user", "content": f"Complete the following sentence in a coherent and meaningful way.\nSentence: {text}\nCompletion: " }
]
```

Based on this template, we attempted various variations and applied in-context learning to improve model performance. We used 1-shot and 3-shot methods, which add 1 or 3 examples, respectively, and kNN- based similar example selection. kNN (k-Nearest Neighbors)-based example selection is a method that selects examples most similar to the input text when selecting examples to include in a prompt. This is because the model's response quality can be improved when examples have greater similarity to the input text. The kNN algorithm calculates the distance between data points and selects the k nearest neighbors, maximizing model performance by providing examples with more similar contexts. Additionally, instruction tuning was used to guide the model to follow instructions better.

3.5 Definition of syntactic complexity

In this study, eight metrics representing syntactic complexity were used to evaluate the performance of an LLM. These metrics were implemented using the spacy, nltk, and benepar libraries, and most of them were derived from previous research on second language learning (L2 learning).

a. Sentence Length

Measures the overall length of a sentence based on the number of words. It is a basic indicator of the overall complexity of a sentence . For example, "She runs" consists of two words, whereas "The quick brown fox that jumped over the lazy dog quickly ran into the forest to escape the hunters" has many more words and is syntactically more complex.

b. Mean Clause Length

The average number of words in each clause. A clause is the smallest sentence unit that contains a subject and a predicate. Similar to sentence length, clause length was added as a metric based on the assumption that sentence complexity increases as they become longer.

c. Mean T-unit Length

T-unit is a sentence unit containing at least one independent clause (subject and predicate) and a subsequent dependent clause. The average number of words in each T-unit in the sentence is calculated. For example, "The cat sleeps | because it is tired." is considered a single T-unit, including the following dependent clause, and is seven words long.

d. Number of Clauses

The number of all clauses (independent clauses and dependent clauses) in a sentence. For example, "She sings and dances | because she is happy." contains three clauses in total: two independent clauses ("She sings and dances") and one dependent clause ("because she is happy").

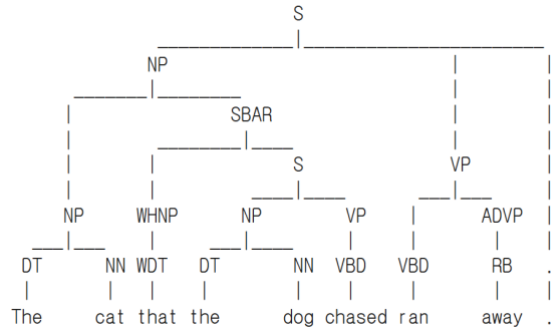
e. Mean Dependents per Clause

The average number of modifiers and adjectives in a clause. For example, "The cat, which is black and white, sleeps peacefully on the mat." is one independent clause, but it contains several modifiers (adjectives and adverbial phrases).

f. Height of the Syntax Tree (Tree Height)

The height of the tree structure when parsing a sentence (parse tree). You can create a parse tree using Python libraries like benepar, spacy, and nltk. For example, the sentence "The cat that the dog chased ran away" is generated as the following tree.

```
(S
(NP
(NP (DT The) (NN cat))
(SBAR
(WHNP (WDT that))
(S (NP (DT the) (NN dog)) (VP (VBD chased)))))
(VP (VBD ran) (ADVP (RB away)))
(. .))
```



g. POS Divergence

Analyze the part-of-speech distribution of words used in a sentence and calculate the standard deviation by counting each part-of-speech (POS).

h. Auxiliary Verb Count

The number of auxiliary verbs used in a sentence. Auxiliary verbs indicate tense, attitude, and state of a sentence. For example, "The cat might have been sleeping" uses three auxiliary verbs—might, have, and been—to form a complex structure.

4 Results Analysis

In this study, we evaluated the performance of an LLM across three tasks: yes/no question answering, text completion, and English-Korean translation. By evaluating the performance of the LLM, we determined the extent to which the LLM could process sentences. Next, we evaluated the correlation between syntactic complexity and model performance to determine which variable best quantifies complexity. Finally, we compared the performance of various prompting methods to determine whether prompting could overcome the limitations of the LLM.

4.1 LLM Performance Verification

gemma-2-2b-it showed an accuracy of 0.718 for boolQ and 0.511 for gardenpath QA. For sentence completion, the perplexity value was 96.077, and the English-Korean translation showed a BLEU score of 0.283.

Overall, the results are not bad, but in the case of gardenpath QA, despite being a binary classification, it showed a performance close to 0.5, which is the performance achieved when randomly selected. Furthermore, compared to the general perplexity of a " good " model, which is usually below 30, the model struggles when selecting the next word.

4.2 Correlation Analysis between Syntactic Complexity and Performance

Next, two approaches were chosen to determine which factors were most significant. The first was a correlation analysis between task-specific evaluations and eight variables . For this purpose, a correlation analysis was performed. In this process, the corr() method was used in the Python pandas library and the correlation coefficient was calculated. This allowed us to determine the degree to which each complexity index correlated with the performance of the LLM. The second was a human evaluation based on syntactic complexity. We classified 150 sentence pairs: 50 "Complex," 50 "Average," and 50 "Simple." Based on these results, we grouped the sentences and applied the same correlation analysis technique to reevaluate the contribution of each indicator to the complexity classification .

(r, p-value)	BoolQ, accuracy	Gardenpath QA, accuracy	Wikitext, perplexity	TED Talks, BLEU score
Sentence length	(0.102, 0.151)	(-0.125, 0.258)	(-0.166, 0.155)	**(-0.088, 0.003)**
Mean Clause Length	(0.070, 0.322)	(-0.090, 0.413)	(-0.180, 0.121)	(-0.021, 0.470)
Mean T-unit Length	(0.108, 0.130)	(-0.119, 0.283)	(-0.176, 0.132)	(-0.026, 0.382)
Number of Clauses	(0.109, 0.125)	(0.069, 0.535)	(-0.019, 0.873)	**(-0.084, 0.004)**
Mean Dependents per Clause	**(- 0.151, 0.033)**	(0.041, 0.709)	(0.066, 0.573)	(0.071, 0.15)
Tree Height	(0.120, 0.090)	(-0.088, 0.426)	(-0.034, 0.769)	**(0.084, 0.004)**
POS Divergence	**(- 0.178, 0.012)**	(-0.066, 0.551)	**(-0.214, 0.0 4 5)**	**(-0.103, 0.00048)**
Auxiliary Verb Count	(-0.086, 0.228)	(nan, nan)	(0.152, 0.194)	(-0.051, 0.082)

▲ Performance – Variable Correlation Analysis Results

4.2.1 Yes/No Question Answer

First, in the case of BoolQ, Mean Dependents per Clause (r = -0.151, p = 0.033) and POS Divergence (r = -0.178, p = 0.012) showed a significant correlation. In contrast, in the case of Gardenpath QA, no significant correlation was found in any indicator. In particular, Auxiliary Verb Count was treated as NaN value in the analysis and thus could not provide the results. This is presumed to be because the Gardenpath QA dataset contains only examples written in a limited sentence structure. In the case of other datasets, the sentence diversity is high and the values of various variables show a wide range, but in the case of Gardenpath QA, the standard deviation values of the variables are very low. The limited dataset size of 84 may also have influenced the low statistical significance.

Next, as a result of grouping, for the BoolQ dataset, POS Divergence showed the most significant correlation across all three groups, and the highest correlation coefficient (r = -0.245, p = 0.005) was observed in the Complex group. Meanwhile, Mean Dependents per Clause also showed a high correlation in the Complex group, indicating that the accuracy of response prediction may fluctuate as the structural details of the sentence increase.

4.2.2 Text Completion

POS Divergence (r = -0.214, p = 0.0 4 5) showed a relatively strong negative correlation compared to other indicators. Other indicators had low or insignificant correlation coefficients.

Next, the grouping results showed that POS Divergence showed a significant negative correlation in all groups, and the result was most prominent in the Complex group (r = -0.291, p = 0.004). Sentence Length and Mean Dependents per Clause showed relatively low correlation coefficients, and the p-values were all over 0.05, so they did not show statistically significant results. It was difficult to see a correlation for the remaining variables as the correlation coefficient values were over 0 or the tendencies between groups were inconsistent. However, in the Average group, tree height showed a negative correlation with performance (r

= -0.112, p = 0.083)

4.2.3 English-Korean Translation

POS Divergence showed the strongest negative correlation (r = -0.103, p= 0.00048) with translation performance (BLEU score). Additionally, sentence length (r = -0.088, p = 0.003) and tree height (r= 0.084, p = 0.004) also showed significant trends.

When the data were grouped, the results were similar to when the entire data was analyzed: POS Divergence showed the strongest negative correlation across all groups and was most pronounced in the Complex group (r = -0.312, p = 0.002).

4.3 Performance comparison according to prompt engineering

4.3.1 Yes /No Question Response

In the BoolQ task, instruction tuning and 1-shot learning achieved identical performance (0.745), demonstrating the effectiveness of concise and effective example-based learning in improving model performance. However, 3-shot learning showed a slight decrease in performance (0.710), suggesting that the large number of examples may have caused unnecessary information overload in the model . KNN-based learning improved performance compared to basic prompting, but underperformed instruction tuning.

In the Garden Path task, instruction tuning achieved the highest performance (0.535), but 1-shot learning significantly degraded performance (0.413). This suggests that the random selection of single examples provided may have confused the model's understanding of the context. In contrast, 3-shot learning and KNN-based learning showed only marginal improvements or maintained similar performance compared to the original prompt.

4.3.2 Text Completion

In text completion, 3-shot learning achieved the best performance, recording the lowest perplexity (36.095). This suggests that providing sufficient examples helped the model effectively understand the context and generate text. Instruction tuning and 1-shot learning also showed significant performance improvements, but KNN-based 3-shot learning recorded a relatively high perplexity (41.372), suggesting that the basic learning method may be more effective.

4.3.3 English-Korean Translation

In English-Korean translation, prompting techniques had a minimal impact on performance. Only 1-shot learning slightly improved the BLEU score (0.031), while 3-shot learning actually decreased performance (0.026). This is consistent with the intuition that translation tasks are more sensitive to data quality and translation rules than to prompts. Instruction tuning and KNN-based learning yielded results comparable to the default prompts.

5 Discussion

5.1 Key findings and implications

This study comprehensively analyzed the impact of syntactic complexity and prompting techniques on the performance of LLMs. The results revealed that POS Divergence was the most significant indicator of performance degradation across multiple tasks. This suggests that the more diverse the parts of speech used in a sentence, the more likely it is that the model will have difficulty understanding the exact meaning and structure of the sentence.

In particular, in the text completion task, POS Divergence recorded the highest

absolute correlation coefficient (-0.214), which is interpreted as the difficulty the model has in predicting the next word in sentences with complex syntax and diverse parts of speech. For example, a sentence like "The committee, which had been deliberating for hours, finally came to an unanimous decision" presents an environment where predicting the next word is difficult due to its diverse parts-of-speech and complex syntactic structure. This increases model uncertainty and degrades the quality of sentence completion.

When grouping data across all three tasks, it was also notable that the Complex group showed the highest absolute value of correlation coefficient between part-of-speech diversity and performance.

Analysis of the impact of various prompting techniques on LLM performance revealed different performance improvement patterns for each task. In Yes/No question-answering tasks such as BoolQ and Garden Path, instruction tuning consistently improved performance, validating the effectiveness of clear and concise command-based prompting. In text completion tasks, 3-shot learning recorded the lowest perplexity, demonstrating an effective approach for text generation through contextual understanding. In English-Korean translation tasks, various prompting techniques did not significantly impact performance, suggesting that the translation model is more sensitive to data quality and translation rules than to prompts .

5.2 Limitations and Further Research Directions

In this study, correlations between variables were not reflected in the analysis. However, there are cases where multiple variables show similar tendencies. For example, the sentence with the highest POS Divergence in the sentence completion dataset is as follows: “Boulter received a favorable review in The Daily Telegraph: " The acting is shatteringly intense, with wired performances from Ben Whishaw (now unrecognisable from his performance as Trevor Nunn's Hamlet), Robert Boulter, Shane Zaza and Fraser Ayres.” Compared to other sentences, both the Sentence Length and the Number of Clauses are big. In this respect, a limitation of this study is that it was not possible to implement a new dataset and confirm the correlation between a single variable and LLM performance while controlling for other variables.

Future research could explore implementing such datasets. This could be accomplished by categorizing sentence structures and artificially creating datasets by replacing words with identical parts of speech. Furthermore, state-of-the-art models like GPT4o could be used to create synthetic data.

6 Conclusion

This study comprehensively analyzed the impact of syntactic complexity and prompt design on the performance of LLMs. The results revealed that POS Divergence significantly impacted performance degradation across multiple tasks, suggesting that models may struggle to handle complex syntax and diverse parts of speech. Furthermore, various prompting techniques demonstrated different performance improvement patterns across tasks, with instruction tuning and 3-shot learning in particular demonstrating significant performance improvements in specific tasks. This study highlights the importance of prompt design for improving LLM performance. Future research could explore more sophisticated correlation analyses by a new dataset.

7 References

Brown, T. et al. (2020). Language Models are Few-Shot Learners. *NeurIPS*.

Carnie, A. (2013). Syntax: A Generative Introduction. Wiley-Blackwell.

Futrell, R. et al. (2020). Comparing GPT-2 and Humans in Processing Garden Path Sentences. ACL.

Lu, X. (2010). A Corpus-Based Evaluation of Syntactic Complexity Measures as Indices of College-Level ESL Writers' Language Development. *TESOL Quarterly.*

Nye, M. et al. (2021). Incremental Comprehension of Garden-Path Sentences by Large Language Models: Semantic Interpretation, Syntactic Re-Analysis, and Attention. *ACL*.

Raffel, C. et al. (2020). A Survey on In-context Learning. arXiv.

8 Supplementary Materials

Performance after grouping – Variable correlation analysis results>

Feature	Complex group (r, p)	Average group (r, p)	Simple group (r, p)
Sentence Length	-0.132, p=0.041	0.105, p=0.056	0.084, p=0.097
Mean Dependents per Clause	-0.189, p=0.012	-0.142, p=0.028	0.101, p=0.045
POS Divergence	-0.245, p=0.005	-0.198, p=0.016	-0.157, p=0.031
Syntax tree height	-0.107, p=0.052	-0.073, p=0.138	0.059, p=0.183

▲ BoolQ

Feature	Complex group (r, p)	Average group (r, p)	Simple group (r, p)
Sentence Length	-0.074, p=0.142	-0.063, p=0.171	0.052, p=0.202
POS Divergence	-0.291, p=0.004	-0.234, p=0.012	-0.201, p=0.026
Mean Dependents per Clause	-0.112, p=0.083	0.097, p=0.102	-0.086, p=0.127
Syntax tree height	-0.067, p=0.195	-0.053, p=0.223	-0.041, p=0.254

▲ Text completion

Feature	Complex group (r, p)	Average group (r, p)	Simple group (r, p)
Sentence Length	-0.105, p=0.031	-0.093, p=0.141	-0.081, p=0.057
POS Divergence	-0.312, p=0.002	-0.254, p=0.006	0.198, p=0.019
Syntax tree height	-0.122, p=0.025	-0.103, p=0.033	-0.091, p=0.045
Mean Dependents per Clause	-0.078, p=0.115	-0.065, p=0.142	0.059, p=0.163

▲ English-Korean translation