



TABLE OF CONTENTS

ABSTRACT	4
ABSTRAK	5
FOREWORD.....	6
TABLE OF CONTENTS.....	8
CHAPTER I INTRODUCTION.....	18
1.1 Research Background	18
1.2 Research Problem	19
1.3 Research Objective.....	20
1.4 Research Scope	20
1.5 Research Advantage.....	21
1.6 Research Schematic	21
CHAPTER II LITERATURE REVIEW	23
2.1 Study on Machine Translation	23
2.2 Study on formalizing natural language	28
CHAPTER III THEORETICAL FRAMEWORK.....	36
3.1 Natural Language Processing.....	36
3.2 Computational Linguistics	36
3.3 Machine Translation	37
3.4 Constituency Parsing.....	38
3.5 Phrase Structure Rule.....	40
3.6 Nanosyntax.....	42
3.6.1 Phrasal Spell-out principle	43
3.6.2 Superset principle.....	44
3.6.3 Spell out driven movement	45



3.6.4 Decomposition of spatial adposition.....	47
3.6.4.1 Path and Place	47
3.6.4.2 Decomposition within Place	48
3.6.4.3 Cognitive underpinnings	50
3.6.4.4 Morphosyntactic Realization	51
3.6.4.5 Semantic Composition	51
3.6.5 WordNet.....	51
3.6.6 Penn Treebank.....	52
3.6.7 Brown Corpus	53
3.6.8 Byte Pair Encoding	53
3.6.9 Dependency parsing	55
3.6.10 Gated Recurrent Unit	57
3.6.11 Graph Convolutional Network.....	59
3.6.12 Attention Layer	60
CHAPTER IV RESEARCH METHODOLOGY	62
4.1 Research Description	62
4.2 Problem Analysis	63
4.3 Research Procedure.....	63
4.3.1 Data Acquisition	63
4.3.2 Lexicon Building.....	75
4.3.3 Implementation	80
4.3.3.1 Lexicalization and machine translation preprocessing	80
4.3.3.2 Vocabulary creation	87
4.3.3.3 Text preprocessing	87
4.3.3.4 Word embedding.....	90



4.4 Model Architecture	91
4.4.1 Model Architecture of Encoder-Decoder with Attention.....	91
4.4.2 Model Architecture of Graph Convolutional Network	95
4.4.3 Evaluation	99
4.4.3.1 Tree validity	99
4.4.3.2 Degree of Syntactic Structure Transformation	99
4.4.3.3 Comparison with distributions before and after transformation	101
4.4.3.4 Sparse categorical cross entropy loss.....	102
4.4.3.5 Accuracy	102
4.4.3.6 Perplexity	103
4.4.3.7 Bilingual Evaluation Understudy (BLEU).....	103
CHAPTER V IMPLEMENTATION	105
5.1 Dataset Acquisition Implementation.....	105
5.2 Environment Preparation	108
5.3 Constructing constituent parse tree	109
5.4 Constructing nanosyntax lexicon	110
5.5 Compiling lexicon entries from various sources.....	111
5.6 Creating lexicon structure and assigning initial values.....	117
5.7 Derivative decomposition of complex prepositional phrases	121
5.8 Text Preprocessing Implementation.....	131
5.8.1 Text Lowercasing.....	131
5.8.2 Text Normalization	131
5.8.3 Make vocabulary	131
5.9 Feature Extraction Implementation.....	132



5.9.1 Building Constituency Tree	132
5.9.2 Tokenization with byte pair encoding.....	136
5.9.3 Building feature with Berkeley neural parser	137
5.10 Model Implementation	137
5.10.1 Model Preparation.....	137
5.10.2 Model Implementation	144
5.10.2.1 Single encoder	144
5.10.2.2 Multi encoder	144
5.11 Model training.....	145
5.12 Evaluation	145
5.12.1 Model Evaluation	145
5.12.2 Nanosyntax lexicon evaluation	147
CHAPTER VI RESULTS AND DISCUSSION	150
6.1 Asian language treebank dataset	150
6.2 Nanosyntax lexicon result	152
6.3 Feature extraction result.....	159
6.4 Model Evaluation	162
CHAPTER VII CONCLUSION AND FUTURE WORK.....	173
7.1 Conclusion	173
7.2 Future Works.....	173
REFERENCES.....	175
APPENDIX.....	180



LIST OF FIGURES

Figure 1 Two parse tree for an ambiguous sentence.....	39
Figure 2 Parse tree of the sentence ‘The book is on the table’ in respect with the toy grammar defined	41
Figure 3 Syntactic structure decomposition in Nanosyntax	43
Figure 4 Illustration of spell out driven movement.....	46
Figure 5 Path and Place phrase decomposition example	47
Figure 6 Decomposition of ‘on top of’	48
Figure 7 Decomposition of ‘in front of’	49
Figure 8 Proposed hierarchy for path, preposition, place, case, and determiner	49
Figure 9 Proposed hierarchy for further decomposition of spatial preposition	50
Figure 10 Minimal Python implementation of Byte Pair Encoding from the dictionary {‘low’, ‘lowest’, ‘newer’, ‘wider’}.....	54
Figure 11 Example sentence with dependency relation illustrating relations: root, nominal subject, object, determiner, compound, nominal modifier, and case	56
Figure 12 Illustration of gated recurrent units. R and z are the reset and update gates, and h and h ⁻ are the activation and the candidate activation	59
Figure 13 Diagram of research outline.....	62
Figure 14 Sample of phrase structure grammar	76
Figure 15 Nanosyntax lexicon for spatial preposition in JSON.....	78
Figure 16 List of Prepositions from English Grammar Book (Kerl, 1861)	79
Figure 17 Workflow of nanosyntax lexicon building	80
Figure 18 Workflow of Implementation stage	86
Figure 19 Illustration of raw text to encoder input	92
Figure 20 Illustration of context word and tree to translation.....	93
Figure 21 Toy example "under the sea" nanosyntax input	96
Figure 22 Toy example "under the sea" adjacency matrix input	97



Figure 23 Graph convolutional network implementation illustration.....	98
Figure 24 Context concatenation as GRU input	99
Figure 25 Distribution of sentence length in English and Indonesian corpus	150
Figure 26 Wordcloud of english and indonesian corpus.....	152
Figure 27 Class distribution in nanosyntax lexicon	153
Figure 28 Constituent tree of the sample sentence.....	156
Figure 29 Lexicalized constituent tree	156
Figure 30 Sample result of accuracy and loss plot on nanosyntax model within 5 epoch	162
Figure 31 Sample result of accuracy and loss plot on nanosyntax model within 10 epoch	163
Figure 32 Sample result of perplexity on nanosyntax model within 5 epoch	164
Figure 33 Sample result of BLEU of nanosyntax model in various sentence length.....	165
Figure 34 Sample of loss trend across models within 5 epoch of training	169
Figure 35 Sample of loss trend across models within 10 epoch of training	170
Figure 36 Sample of perplexity trend across models within 5 epoch of training	171
Figure 37 Sample of perplexity trend across models within 10 epoch of training	171
Figure 38 Sample of BLEU scores across models by sentence length ...	172



LIST OF TABLES

Table 1 Literature Review of Machine Translation	26
Table 2 Literature Review of Natural Language Formalization	34
Table 3 Dependency labels from Universal Dependency project (De Marneffe et al., 2021).....	55
Table 4 Example of the dependency relations	56
Table 5 Sample of entries in Treebank split of Asian Language Treebank	65
Table 6 Sample of entries in parallel corpus of Asian Language Treebank	67
Table 7 Sample of sentences in REAL Corpus	74
Table 8 Sample of sentences in ReferItGame	75
Table 9 Illustration of Spatial Decomposition and Substitution	82
Table 10 Illustration of other machine translation preprocessing as comparison to baseline.....	84
Table 11 Toy vocabulary example	87
Table 12 Text with unicode standarization step.....	88
Table 13 Text with lowercasing step	88
Table 14 Text with character filtering step	89
Table 15 Text with punctuation separation step	89
Table 16 Adding text with start and end tokens step	89
Table 17 Text to index step.....	90
Table 18 Index to embedding vector table lookup.....	91
Table 19 Architecture of encoder decoder with attention model concatenated with GCN with layers, output shape, parameters, and connected layer	94
Table 20 Classification of preposition by Svenonious.....	110
Table 21 Sentence statistics of asian language treebank dataset.....	150
Table 22 Type to token ratio (TTR) and out of vocabulary (OOV) corpus analysis.....	151
Table 23 Nanosyntax lexicon entries of “above”.....	154



Table 24 Nanosyntax lexicon entries of “into”	154
Table 25 Nanosyntax lexicon entries of "close"	155
Table 26 Tree edit distance result	158
Table 27 Part-of-speech tag distribution before and after lexicalization	159
Table 28 Samples of output from various feature extraction as machine translation preprocessing.....	161
Table 29 Translation sample 1 of nanosyntax model.....	166
Table 30 Translation sample 2 of nanosyntax model.....	166
Table 31 Translation sample 3 of nanosyntax model.....	167
Table 32 Machine translation training and inference result.....	168



LIST OF SOURCE CODE

Source Code 1 Importing treebank data split Asian Language Treebank with pandas library from hugging face.....	105
Source Code 2 Importing parallel corpus of Asian Language Treebank	106
Source Code 3 Merge treebank and parallel corpus to get complete corpus alignment.....	106
Source Code 4 Convert ReferItGame dataset from dictionary to dataframe	107
Source Code 5 Imported libraries for building lexicon and machine translation	109
Source Code 6 Constituent tree generation process	110
Source Code 7 Merging and compiling semantic relation preposition from english book and Wikipedia	113
Source Code 8 Querying preposition keyword definition in Merriam Webster online dictionary	113
Source Code 9 Scrapping definition from Meriam Webster online dictionary ..	115
Source Code 10 Decomposing current preposition lexicon with substring checking with WordNet.....	117
Source Code 11 Filling atomic tokens as entries in the lexicon	119
Source Code 12 Compiling list of non-atomic prepositions from Wikipedia.....	121
Source Code 13 Initialize master lexicon process.....	122
Source Code 14 Add wordnet prepositions to complex preposition list.....	122
Source Code 15 Complex preposition decomposition implementation.....	123
Source Code 16 Initialization of Factory class and main sequence of decomposition	124
Source Code 17 Segment data extraction and spell out head processing	126
Source Code 18 Role refinement and entry construction utilities.....	127
Source Code 19 Path phrase and Place phrase classification logic.....	128
Source Code 20 Dynamic class creation from preposition phrase (PP) list.....	128
Source Code 21 Individual PP tokenization and syntactic tree construction.....	130
Source Code 22 Text lowercasing process.....	131
Source Code 23 Text normalization process.....	131
Source Code 24 Building source and target language vocabulary process.....	132



Source Code 25 Parser initialization and constituency tree preprocessing.....	132
Source Code 26 Constituency parser class and vocabulary construction	133
Source Code 27 Tree to graph conversion with adjacency and feature matrix process	135
Source Code 28 Graph data with edge index process	136
Source Code 29 Initialize pretrained byte pair tokenizer model.....	136
Source Code 30 Constituency parsing process	137
Source Code 31 GRU Encoder and Decoder with attention	139
Source Code 32 Decoder inference methods	140
Source Code 33 Graph Convolutional Network Layer	141
Source Code 34 Multi Encoder with Attention decoder for tree and text input .	143
Source Code 35 Encoder with Attention decoder initialization.....	144
Source Code 36 Multi Encoder with Attention decoder initialization.....	145
Source Code 37 Translation model training process implementation	145
Source Code 38 Masked perplexity implementation	146
Source Code 39 Compile model with optimizer and metric	146
Source Code 40 Calculate BLEU process.....	147
Source Code 41 Calculate Tree Edit Distance process	148
Source Code 42 Calculate distribution distance with chisquare	148
Source Code 43 Compare distribution of lexicalized syntax	149