Master Thesis

Basic Level Detection: Learning from natural language corpora with various characteristics and synthetic features

Author: Haochen Wang (VU: 2698251 UvA: 13500198)

Internal supervisor: Adam S.Z. Belloum (UvA)
External supervisor: Laura Hollink (HCDA, CWI)

A thesis submitted in fulfillment of the requirements for the joint UvA-VU Master of Science degree in Computer Science

July 18, 2022

"As celestial bodies maintain vigor through movement, a gentleman should pursue constant self-improvement"

from I Ching

Day of the defense: July 13, 2022

Abstract

Basic level, a psychological term, the core idea is that in a hierarchy of concepts there is one level of abstraction that has special significance for humans to perform and react quicker and more accurately than at superordinate or subordinate levels. The master thesis aims at detecting the basic level in a hierarchy. The goal is to come up with a relatively reliable method to predict the basic level concepts from natural language corpora and synthetic features. The synthetic features can be from the lexical and hierarchical structure, frequency in corpora, and semantics from word embeddings and generated features. Using knowledge and techniques in machine learning, natural language processing, and statistics, the thesis answers the research question that *to what extent the basic level can be learned from natural language corpora with various characteristics and synthetic features.* This thesis proposes a method to predict basic level concepts in a hierarchy. We find the features of large size of written corpora to general audiences can enhance the performance of predicting the basic level. Using the method and the best setting of the features, we reach the best Cohen's kappa of 0.898 to predict the basic level and create a dataset of predicted basic level concepts in WordNet under *entity.n.01.*

Keywords: *Basic-level categories, Machine learning, Natural language processing, Semantics, Psycholinguistics*

Acknowledgements

This study was carried out under my traineeship for master's graduation at Human-Centered Data Analytics Group, Centrum Wiskunde & Informatica. I would like to express my sincere gratitude to my thesis supervisor Dr. Laura Hollink, group leader of Human-Centered Data Analytics, CWI, for her patient guidance, enthusiastic encouragement, and useful critiques of this research project. I would like to extend my thanks to group members in HCDA, Dr. Lynda Hardman, Dr. Davide Ceolin, Dr. Dirk Meijer, Savvina Daniil, Andrei Nesterov, Boyu Xu, Julia Sydnik, and Ghazaleh Tanhaei. They gave me helpful feedback and advice on the presentation and writing. I would also like to show gratitude to supervisor Dr. Adam S.Z. Belloum, UvA, for his patience, guidance, and valuable feedback on the project.

Last but not least, I am grateful to my family whose constant love and support keep me motivated and confident. Also, my warm thanks go to my friends and fellows.

Contents

List of Figures

List of Tables

1

Introduction

Basic level, also known as basic-level categories, is introduced by Rosch et al. in 1976. It is a terminology in psycholinguistic literature. The basic level is the most general categories for which a concrete image can be formed. And it is hypothesized that the definition of concrete categories during child development is based on semantic features of each individual in the child's perception (5). The core idea is that in a hierarchy of concepts there is one level of abstraction that has special significance for humans to perform and react quicker and more accurately than at superordinate or subordinate levels.

Knowledge organization systems (KOS) are widely utilized by applications on the web. They organize and provide an abundance of information for applications. KOS requires the categorization of concepts and explicit semantic relationships among them to represent the knowledge. Especially for Semantic Web and its technologies, it is useful to turn to cognitive models and theories to replicate human thoughts to categorize concepts on the web (6). The basic level theory is one of the best categories for both human understanding and KOS organizing.

Research and applications derived from the basic level are rich and promising. Applications using knowledge bases could improve their user interfaces if they adopt appropriate basic-level categories. More exactly, if applications can detect concepts belonging to the basic level, they would provide more efficient service and a better user interface. Moreover, many methods in modern NLP and computer vision research have turned out that the basic level is useful to improve performance. Some published research includes ontology generation (7), word sense disambiguation (8), and visual object recognition (9).

However, the concepts belonging to the basic level are not labeled originally, and there lacks a general and robust method to detect the basic level categories from knowledge bases. Therefore, this project aims at the topic of basic level detection. The goal is to

come up with a relatively reliable method to predict the basic level concepts from corpora with various characteristics and synthetic features. The corpus characteristics contain the discourse type, target audience, and size of a corpus. The synthetic features include structural, frequency, and semantic features.

In overview, basic level detection is a classification task in machine learning. Because basic level concepts usually take only a small part of all the synsets, this is an imbalanced classification task. We consider machine learning algorithms, Random Forest with the SMOTE algorithm and Support Vector Machine. Some features for the machine learning algorithm are inherited from Henry's (10), and newly generated by techniques from natural language processing. Frequency could be an important feature to predict the basic level. People often choose basic level concepts to describe an object and the basic level should be frequent in the language (5)(1). Google Books Ngram Corpus as a source is used to extract the frequency features. The semantic features are from Word2vec and BART. The features used for training the classifier are aggregated and selected by feature engineering. The statistic tests could give significance to the hypotheses which provide convincing evidence to make conclusions. Wilcoxon rank-sum test, a non-parametric statistical test, is carried out. Finally, we use the synthetic features with the best performance setting to detect the basic level in a large-scale hierarchy.

In this thesis, we investigate the performance of corpus characteristics and the synthetic features in predicting the basic level. An end-to-end method from a concept in a hierarchy to its label whether it is the basic level is proposed. Using this method, we detect basic level concepts under *entity.n.01* in WordNet. The codes, the related corpora, the generated cues of concepts, and the predicted basic level concepts are made available in GitHub[1].

The main research question is that **to what extent the basic level can be learned from natural language corpora with various characteristics and synthetic features?** To be specific, three sub-research questions will be answered:

- To what extent the discourse type and target audience of a corpus considering its size would affect the performance of predicting the basic level?
- What new features concerning semantics can be generated to help improve the performance of predicting the basic level?
- How much would corpus characteristics and synthetic features improve basic level detection in a large-scale hierarchy?

[1] `https://github.com/DanferWang/Basic_Level_work`

2

Background

2.1 Literature Study

2.1.1 Basic Level Categories Theory

Basic level as a level of abstraction in taxonomy is observed in 1958 by psychologist Roger Brown. He postulated that there exists a preferred level of names that is the most useful in most contexts (11). However, Brown did not clearly define this level nor give a description of the names that can belong to it. A formal name for basic-level categories and a systematic theory of basic-level categories are defined by psychologist Eleanor Rosch in 1976 (5). Research on the basic-level categories has been conducted across diverse disciplines. Studies in psychology, anthropology, linguistics, and library and information science have more or less covered the theory of basic-level categories to measure perception, communication, and behavior (12).

Besides Brown and Rosch, linguist George Lakoff raised a research question that aims to study the human mind through the categories of language. He demonstrates that basic level categories are 'human-sized' and depend upon human interactions with objects in a category (13). Rebecca Green stated in the field of library and information science that concepts in basic-level categories have been demonstrated to have more possibilities to be shared across classificatory systems than others by (14).

Although there are many publications on basic level categories, they hardly give a specific mathematical definition to the basic-level categories. Most of the related work describes the basic level categories theory with an intuitive idea, such as categories containing the most information or people would react fast to these categories. In our work, we give a formal definition of the basic-level categories from semantic and quantitative aspects. The semantic one is the cognitive economy which explains that people tend to make less effort

to understand and react to the basic-level categories. The quantified one is cue validity (5) which describes how much information is gained by people from the basic-level categories. In this way, basic-level categories, a terminology of psychology, can not only be described by a level of abstraction, but measured by a mathematical formula as well. The cognitive economy and the cue validity are essentially the same and used to define the basic-level categories.

Some terminologies about basic level categories are similar to each other. The basic level is a level at which concepts belong to basic-level categories. Therefore, basic level and basic-level categories have the same meaning in this paper.

2.1.1.1 Cognitive Economy

Humans perceive the real world with correlational structures. The perception is from cognitive processes which tend to minimize exertion and resources cost during processing. Further, cognitive economy concerns the relevance and simplicity of a categorization scheme and knowledge representation (15). From the aspect of cognitive economy, the basic level, as a criterion of binary classification, can result from a combination of the following two principles (5).

1. **Predictable property.** Concepts in the basic level would have many predictable properties which can be derived from each other. The attribute of predictable properties leads to forming a large number of categories. There are discriminations in each category and one of them holds fine enough differences to distinguish each concept belonging to it. This category is likely to be the basic level.

2. **Relevant differentiation.** The categorization scheme aims to reduce infinite differences to an appropriate degree of proportions among concepts within one category. The appropriate degree of proportions behaviorally and cognitively depends on the purposes. Otherwise, it would not differentiate among concepts unless the differentiation is relevant enough with respect to the purposes.

The two principles look contradictory to how humans perceive the world with categorization. They emphasize an appropriate degree of differentiation according to human interaction with the world under different situations. For example, assuming a concept hierarchy (1) shown in Figure 2.1, the concept of cat and the concept of dog can be in the basic level according to the principles of cognitive economy. In the hierarchy, concepts higher than cat and dog are more abstract whose properties can not be the same

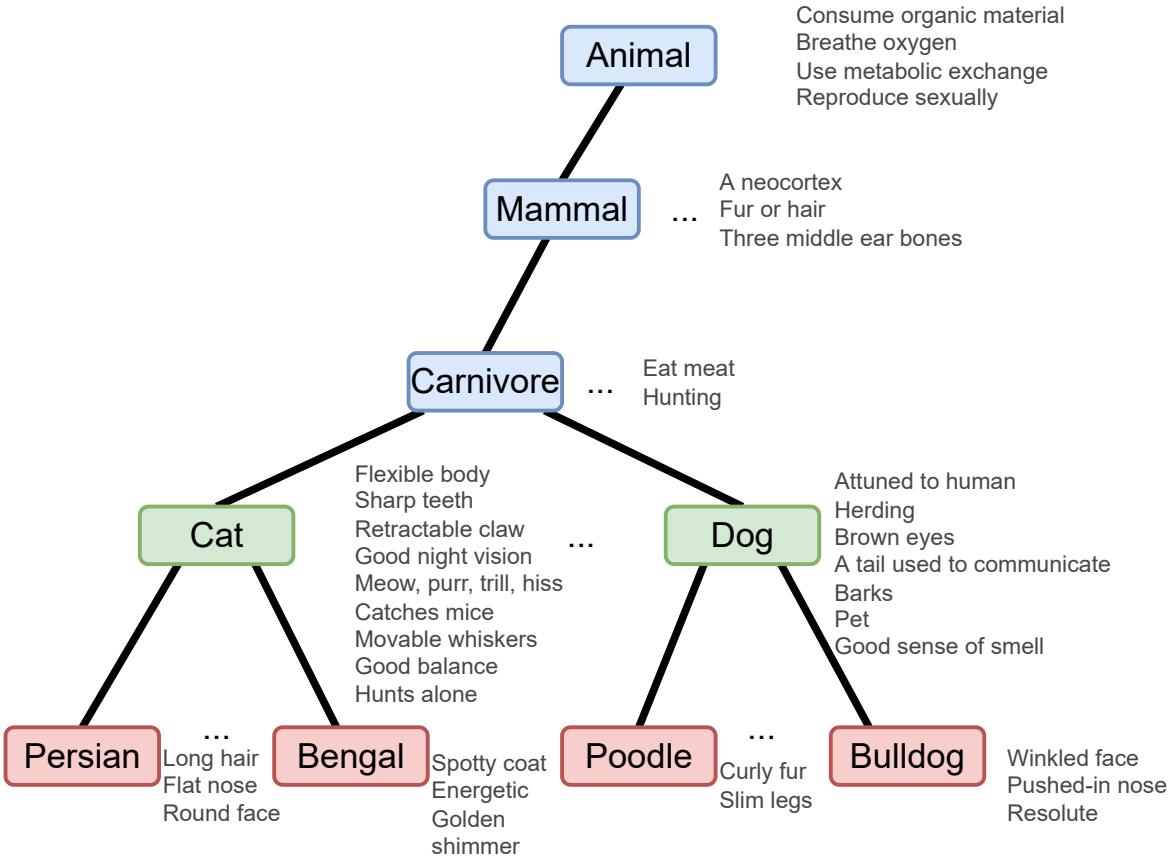

Figure 2.1: Concept hierarchy with properties example (1)

as predictable as cat and dog. Similarly, concepts lower than them are too specific whose properties indicate only slight differentiation. The concept of cat and the concept of dog both hold as many predictable properties as possible to distinguish them from the others (Predictable property). Meanwhile, a category of the concept of cat and the concept of dog can be the most appropriate proportions of differentiation for a general perception by humans (Relevant differentiation).

By basic-level categories, concepts more abstract or general than those in basic level are superordinate concepts, i.e. hypernyms in a hierarchy. Concepts more specific than or below the basic level are subordinate, i.e. hyponyms in a hierarchy. With the basic level theory, humans can sketch the real-world correlational structures.

2.1.1.2 Cue Validity

Cue validity, $val(cue)$, is based on conditional probabilities which typically include $P(BL|cue)$ and $P(\overline{BL}|cue)$ terms. $P(BL|cue)$ is the probability of a concept is **IN** basic level given the cue, while $P(\overline{BL}|cue)$ is the probability of a concept is **NOT IN** basic level given the cue. Qualitatively, $val(cue)$ goes up when $P(BL|cue)$ increases and(or) $P(\overline{BL}|cue)$ decreases. However, there is no commonly agreed upon mathematical formula to calculate the cue validity. BEACH proposed probabilistic cues (Equation 2.1) to make inferences about objects' category (16). Here, the object can be seen the same as the concept in this paper.

$$E(k) = \frac{\sum_{d=1}^{n} P(k|d)}{n} \tag{2.1}$$

Where k is a specific category that one object probably belongs to. $E(k)$ can be regarded as a possibility of one object correctly expected to be in a given category. d is a dimension on which one object's cue is known. n is the number of dimensions. $P(k|d)$ is the relative frequency with which one object's cue on each cue dimension (16). Beach improved the inference method in another paper. Another formula (Equation 2.2) was put forwards to recognize, assimilate, and identify a category for an object (17).

$$E(c) = \sum_{d=1}^{n} \frac{P(c|k,d)}{n} \tag{2.2}$$

Where c is a cue value under consideration as one object's unknown cue. $E(c)$ is the total evidence from the unknown cue dimension. $P(c|k,d)$ is a probability that a cue value c on an unknown cue dimension d is the best bet for the inference give one object's category k (17). Based on Beach's study, Reed updated the algorithm to calculate the cue validity and proposed a similar formula, Equation 2.3, which measures the cue validity by considering the frequency and the proportion of cues in categories (18).

$$CV(Category\_k, X) = \sum_{m=1}^{d} \frac{P(Category\_k|x_m)}{d} \tag{2.3}$$

Where k is an ordinal number of categories while m is an ordinal number of cues of X. $CV(Category\_k, X)$ is the cue validity value of concept X. $P(Category\_k|x_m)$ is the prior probability by $P = 1/(1+F)$, where F is the frequency with which the cue appears in the category (18).

According to the definitions of basic level, a concept with a larger cue validity can be more differentiated than others with a lower one. It is reasonable that the superordinate concepts have fewer attributes in common to have lower cue validities. Meanwhile, the

subordinate concepts share many attributes among siblings that lead to lower cue validities. Concepts in the basic level maximize cue validity, in other words, these concepts reflect the correlational structure of the real-world environment best and are identified fast for humans.

When computing cue validity of a concept, the practical implementation does not lead to Equation 4.2 because it contains a posterior probability that is impossible to count and calculate for training and testing. The detail of implementing cue validity of concepts are discussed in Section 4.4.2.

2.1.2 Random Forest with SMOTE

Basic level detection is to categorize a concept into basic level or non-basic level which is a binary classification task in Machine Learning. Random Forest (19) with Synthetic Minority Over-sampling Technique(SMOTE) (20) will be used as a classifier to learn from synthetic features and to predict concepts whether are in the basic level or not.

Random Forest is an extensive version of Bagging, which is the abbreviation of Bootstrap Aggregating. It uses Decision Tree as a base learner and builds a Bagging aggregation. Furthermore, Random Forest introduces the random choice of attributes in the process of training. The core concept of the fundamental method, Bagging, is sampling and training for every subset of the attributes. They can be sampled for training a Decision Tree with a certain number of items. The training leads to a base learner. Hence, several base learners can be integrated or aggregated into a final Random Forest learning model.

Specifically, training a Random Forest classifier includes sampling attributes and choosing an attribute. Firstly, a subset of the total d attributes is sampled for each current node in the base Decision Tree in a bootstrap way. The size of every subset is k. Secondly, the most optimal attributes from the subsets are chosen to generate their child nodes respectively. In this way, the parameter k controls the degree of randomness introduced. In general, according to (19), the recommended value would be:

$$k = \log_2 d \tag{2.4}$$

Random Forest is relatively simple to implement and needs low computational cost. Moreover, it has shown powerful abilities and performance in many Machine Learning tasks. Both the self-sample perturbation and the self-attribute perturbation enable better performance of generalization via increasing the bias of individual base learners. Therefore, Random Forest usually converges to lower generalization errors with the increment of the number of base learners.

When training a Random Forest with imbalanced datasets, SMOTE algorithm for sampling can achieve better classification performance. In general, SMOTE is a method that oversample minority classes and undersample majority classes (20).

When undersampling the majority classes, the samples are removed at random until the percentage of samples in the majority classes and the minority classes reaches a specified value. When oversampling the minority classes, besides taking minority samples, it creates synthetic examples of each minority instead of directly replicating with replacement. The synthetic examples are calculated with the K-Nearest Neighbor algorithm. Discovering the nearest neighbors of a minority sample, each difference between the minority and its nearest neighbors is multiplied by a random number ranging from 0 to 1. Then, the new examples can be added to the dataset for training which leads to a classifier having greater decision regions but less specific than without such oversampling.

2.1.3 Word Embedding: Word2vec

Word embedding is an important technique in natural language processing that words are mapped to vectors of real numbers. It is a necessary procedure in modeling a language and learning features from textual data to numerical representations. Word embedding aims to capture the meaning of a word in semantic similarity, syntactic similarity, and relations with other words which makes natural language read and processed by computers. A well-trained set of word embeddings place similar words close to each other in the vector space. Based on the real-number representation of words, further computation and algorithms can be implemented.

Word2vec (21) is one of the most popular techniques to learn word embeddings using multi-layer recurrent neural networks. There are two main training algorithms for Word2vec, the continuous bag-of-words(CBOW) model and the skip-gram model. The major difference between the two models is that CBOW uses context to predict a target word while skip-gram uses a word to predict the target context. According to Mikolov (22), the skip-gram model is more suitable for representing not frequent words. To hit lemmas of the concepts in the dataset Section 3.2 as many as possible, a Word2vec-based repository named *ConceptNet Numberbatch*[1] can provide finely pre-computed word embeddings trained with data from *ConceptNet* (23) using the skip-gram model. Compared to other pre-computed word embeddings, *ConceptNet Numberbatch* is able to cover most of the lemmas in our dataset which guarantees to eliminate missing vectors as few as possible.

[1] `https://github.com/commonsense/conceptnet-numberbatch`

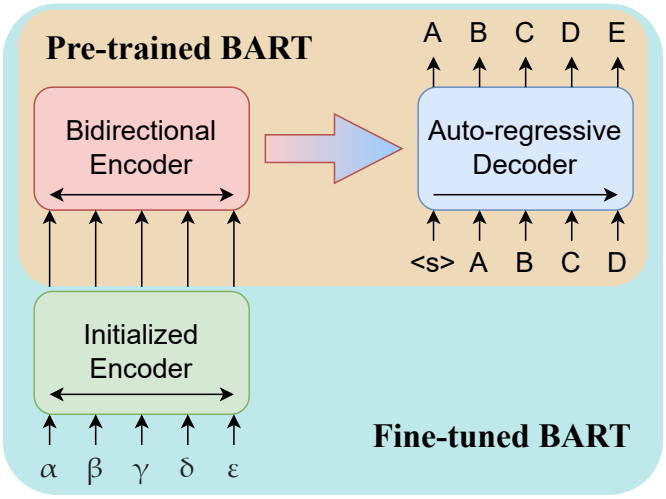

Figure 2.2: Architecture of BART (2)

The vector representation with the semantics of a lemma of concepts in a hierarchy can be looked up in the dictionary of *ConceptNet Numberbatch*.

2.1.4 BART

BART, Bidirectional and Auto-Regressive Transformers, is a sequence-to-sequence denoising autoencoder model (2). It is one of the effective language models for text generation and comprehension tasks, such as machine translation. By fine-tuning BART, an end-to-end model can be trained which can learn a mapping from source English words to their semantic features. In this paper, BART is used as a pre-trained model for tokenization and a fine-tuned autoencoder for semantic feature generation.

The architecture of BART follows the standard Transformer (24). It is implemented with a bidirectional encoder and an auto-regressive decoder, shown in the yellow area of Figure 2.2. The pre-trained BART is denoising because the input training data is corrupted text with masks(attention) and the goal is to reconstruct the text which is noticed by the masks. Both the encoder stacks and decoder stacks contain 12 identical layers (24). The encoder of the pre-trained BART then can be used as a tokenizer for English words. It gives a vector of identity with an attention mask to each word or phrase which implicitly represents the meaning.

After fine-tuning the BART with an additional encoder, named Initialized Encoder, the new model is designed for machine translation tasks, shown in the green area of Figure 2.2. The pre-trained BART without the embedding layer is used as a decoder. The new encoder

is trained to map the input source text into an intermediate representation which can be denoised by the pre-trained BART (2). The BART is fine-tuned with English semantic feature data, to be introduced in Section 3.5. Using the fine-tuned BART, the end-to-end model of translation can help to generate semantic features from a word. In other words, the translation is a mapping from one concept to its semantic features. The pipeline and detailed process of fine-tuning will be discussed in Section 4.4.2.

2.2 Related Work

2.2.1 Rule-Based Heuristics

Mills et al.(25) built a rule-based system with heuristics to identify basic level categories automatically. Their approach is to evaluate a cumulative set of rules defined by themselves. The system constraints concepts are the basic level with some boundaries of the rules. Initially, there are 52 rules in two types: filtering rules and voting rules. They used several resources of corpora, dictionaries, and toolkit to formulate the rules. After experiments of training and developing, there are 8 chosen filtering rules with parameters and 4 selected voting rules left for relaxation using a greedy search scheme.

Although the system can identify the basic level with relatively high accuracy of 77.0% and classify automatically, the data gathered was limited, 194 categories in total. For the reason that some categories do not have corresponding synsets in WordNet, the categories used in the experiment are even fewer, 152. Moreover, there could be many important features ignored because of the removal of weak rules. It might not work well with concepts outside the 152 categories because the rule-based system is trained and developed with only 100 categories. In our work, experiments are conducted with more annotated concepts, up to 839, and have different models designed to guarantee the generalization of the method for predicting the basic level.

2.2.2 Machine Learning-Based Classification

Recently, more related research on predicting the basic level focuses on Machine Learning. Concepts can be categorized into the basic level or others using several kinds of classifiers. With Machine Learning algorithms, predicting the basic level is regarded as a classification task. Moreover, appropriate feature engineering can improve the accuracy and efficiency of predicting.

Hollink et al. (26) aim to predict whether concepts are the basic level in a concept hierarchy. They trained five kinds of classifiers from three types of features: lexical features,

structural features, and frequency features. The classifiers are trained by Latent Dirichlet Allocation(LDA), Decision Tree, K-Nearest Neighbors, Support Vector Machine(SVM), and Random Forest. The lexical and structural features are extracted from WordNet (3), while the frequency is from Google Books Ngram Corpus. They present a method to classify concepts from a conceptual hierarchy into a basic level and non-basic level using Random Forest. The models are trained in the setting within one domain and across domains. The local model, whose training data is within a domain, results in the best performance under three domains. They argue that concepts that are difficult to label for humans are also harder to classify automatically.

The method Hollink et al. considered and the features they chose concern the structure of concepts in a hierarchy and their lemmas morphology. The lexical features and structural features implicitly contain some semantic relations among synsets from their hypernyms and hyponyms. The implicit semantic relations could indicate the subordinate relationship, however, might not be able to summarize the meanings of one concept(synset). In our work, the semantics of concepts is explicitly represented by their cues generated by the fine-tuned BART. The cues of a concept could explain the synset directly rather than inferred from the subordinate relationship. The method proposed in our project adopts both the implicit semantics from the hierarchy and the explicit semantic features by cues.

Henry (10) focuses on the features from corpora. She raises a research question that what corpus properties are useful in predicting the basic level. It is through learning the basic level with varying corpora of different discourse types, audience ages, and sizes in words. She concludes that larger corpus sizes have more reliable results. And comparing smaller samples of the same size, those containing spoken discourse and discourse directed at children provided more reliable results than written text aimed at a general audience. The features from child spoken corpora can be important indicators to learn and predict the basic level.

It reveals the significance of the type and the size of frequency sources. However, the aggregations of frequency features from different corpora are not the same. The performance of accuracy for predicting the basic level is not improved significantly from these frequency features. Compared to our work, Henry did not consider semantic features of concepts.

Chen and Teufel (1) present the first method for the detection of the basic level at scale using Roach-style semantic features which contain cue validity, according to their statement. They adopt three methods of generating semantic features for synsets in WordNet: textual features from Wikipedia pages, Distributional Memory (27), and BART. The languages are English and Mandarin. The synthetic textual features include structural

features, lexical features, Word2Vec, frequency features, cue validity, basic level page rank, and semantic features. Support Vector Machine is used to train the classifier. We also use BART and the cue validity but for different purposes. They used BART to compile the synthetic texts for a better Jacquard Distance (28) and to tackle multi-language issues. BART in our method is used to fine-tune a generator for cues of concepts. Cue validity in Chen and Teufel's was a feature-dependent indicator for checking the quality of the synthetic features (1). In our work, cue validity is used to extract the semantic feature from generated cues and represent it in a numerical way.

Although Chen and Teufel find that BART is capable of generating indicators to improve the detection of the basic level, they might not clarify the mechanism of BART nor the functionality of the generation. The best model in their experiments performs 75.0% accuracy of English basic level detection and 80.7% in Mandarin on their test set. However, the dataset only contains 433 concepts which are carefully selected and not directly from a developed hierarchy.

2.2.3 Context-Aware Basic Level in Folksonomies

Chen et al.(7) put forward an algorithm to detect the basic level among various contexts from folksonomies. The folksonomies contain implicit semantics from creating and managing tags in web resources annotated by users. They model instances, concepts, and context in the folksonomies for mining semantics. Contextual category utility, inspired from category utility(29), is proposed to predict the basic level. The modeled concepts are detected as the basic level when they have the greatest value of the contextual category utility.

Chen et al. considered semantics when predicting the basic level and with large-scale web resources. Basic level concepts are detected on the web where the concepts are not organized hierarchically. The results depend on the contents and quality of the resources. Their method is appropriate for detecting the basic level within an abundant source of texts. However, it might be not suitable in a hierarchy because the folksonomies might not cover all the synsets in a hierarchy, like WordNet. Moreover, due to the context-ware method, there is no use of features from lexical, frequency, or structural characteristics.

3

Data

3.1 WordNet

Princeton WordNet[1] is a lexical database for English which organizes sets of synonyms (3). The sets of synonyms, known as *synsets*, are regarded as concepts in this paper. The canonical form or morphological form of a word from the synonyms is one of the *lemmas* of the synset. The meaning of each word is named *sense*. By these definitions, the lexical semantics can be described in terms of the relations by the links between the synsets. There are over 166,000 relations, which are represented in pairs of a lemma and a sense, and more than 117,000 *synsets* in WordNet (3).

Another important semantic relations are hyponymy and hypernymy which are the transitive relations between *synsets*. *Hyponyms* and *hypernym* can shape paths from the superordinate to several Subordinate. Therefore, concepts in WordNet are organized in a hierarchical structure of lexicons. Further, every synset together with its *hyponyms* and their relations contain the lexical information in the hierarchy, which is used as one of the lexical features. A hierarchy of the concepts in our annotation dataset is depicted in Figure 3.1. The synsets in light red are the domains in the dataset Section 3.2. Synsets in yellow are the hyponyms of the domains and synsets in blue are their hypernyms. The root of the hierarchy is the synset of *entity.n.01*.

As mentioned in Section 3.2, concepts to be predict can be the *synsets* in WordNet. Moreover, the method proposed can be executed to detect the basic level with all the concepts under *entity.n.01*. WordNet database and its API can be accessed by NLTK[2] WordNet Interface[3].

[1] `https://wordnet.princeton.edu/`

[2] NLTK: Natural Language Toolkit

[3] `https://www.nltk.org/howto/wordnet.html`

Domain	Basic level	Non-basic level	Total
hand tool	25	108	133
edible fruit	57	99	156
musical instrument	47	79	126
furniture	20	163	183
garment	26	215	241
Total	175	664	839

Table 3.1: Summary of Basic Level Annotation Dataset

3.2 Basic Level Annotations

The dataset where concepts are labeled with basic level or non-basic level is inherited from Hollink et al. (26) and Henry's research. There are three domains from Hollink's dataset and two domains from Henry's. The domains are *hand_ tool.n.01*, *edible_ fruit.n.01*, *musical_ instrument.n.01*, *furniture.n.01*, and *garment.n.01* in WordNet. The labeled dataset is called the gold standard. Originally, the gold standard labels concepts in the basic level, or the superordinate or the subordinate of the basic level. In this paper, superordinates and subordinates are merged into a class of the non-basic level.

Concepts in the gold standard are labeled manually by three annotators who are provided with an annotation protocol. The protocol includes instructions for this labeling task, descriptions of the basic level, characteristics of the basic level, and how to find the basic level in the hierarchy of WordNet. The most important part is a checklist helping label the basic level. In addition to the checklist, the annotators may access necessary information from Wikipedia and Google Search Engine. Using the annotation protocol, concepts labeled as the basic level can be as close as possible to Roach's definition of the basic level, discussed in Section 2.1.1.

After processing the gold standard, the dataset to be used in experiments is summarized in Table 3.1. It shows distributions of the number of concepts in each domain. The data is imbalanced that concepts at the non-basic level are $2 \sim 8$ times more than those at the basic level. Considering the definition of the basic level, it is reasonable that the basic level is less but contains more information in one domain. The settings of the training set, validation set, and testing set will be discussed in Section 4.1.

3.3 Textual Corpora

To answer the first research question about frequency features, different resources for calculating frequencies of concepts should be considered. The aim is to gather the frequency of lemmas in each concept from different discourse types, different target audiences, and various sizes of resources. Therefore, four corpora with different characteristics are extracted to be used as the frequency resources. Summarized in Table 3.2, they are the KBNC, the CABank English corpus(CABNC) (30), the CHILDES (31), and the British National Corpus(BNC) (32),

Text BNC is a British English corpus that contains around 100 million words from a wide range of written and spoken resources. It records abundant British English from the late 20th century and is released in 2007 as *BNC XML Edition*[1]. Approximate 88 million words of written records are extracted and marked as BNC Written corpus for the frequency feature under a general written corpus. Meanwhile, there are around 1 million records of them specifically for children. They are wrapped as KBNC which is a written corpus specific to children.

CABNC is built by re-transcribing naturalistic conversations from Audio BNC, a subcorpus of BNC which originally contains about 7.5 million words in a type of audio. Albert et al. converted the transcripts into CHAT files (33) and made them public open-licensed[2]. CABNC initially has around 4.2 million words. However, from the latest version released only 2.4 million words can be parsed from CHAT files by *PyLangAcq*[3]. The parsed words compose CABNC for calculating frequencies of concepts under a general spoken corpus.

CHILDES is one of the components in the TalkBank system specific for child languages[4]. 16 corpora from British English consist of a new corpus, simply named CHILDES. The new CHILDES contains conversations directed at children with around 5.7 million words that are transcribed from audio and video. It is used to extract frequency features as the corpus characteristics of spoken discourse specific to child audiences.

3.4 Google Books Ngram Corpus

To have a larger corpus for extracting the frequency features, Google Books Ngram Corpus(Google Ngram) (34) can be another resource that is a written corpus for general

[1] `http://www.natcorp.ox.ac.uk/`
[2] `https://ca.talkbank.org/access/CABNC.html`
[3] `https://pylangacq.org/`
[4] `https://childes.talkbank.org/`

Corpus	Discourse Type	Target Audience	Size Approx.	Description
KBNC	Written	Children	1 million	Subset of the BNC specific for children target audience
CABNC	Spoken	General	2.4 million	Re-transcribed from a subcorpus of the BNC
CHILDES	Spoken	Children	5.7 million	Composed of 16 sub-corpora
BNC	Written	General	88 million	British English in the late 20th century

Table 3.2: Summary of Corpora for frequency features

audiences. It is an enormous repository of printed publications. Similar to Hollink et al. and Henry's study, frequency features from Google Ngram could be a set of important indicators for the classification. The corpus has three versions. We adopt in our work the third version released in 2020. It contains millions of books published since the 1500s. Although the accuracy number of tokens in the Google Ngram version 3 is not documented, it can be sure that the amount is larger than the second version, which is over 468 billion tokens (35). And it was updated by billions of records annually from 2012 to 2019. The n-gram data used is all the entries in Google Ngram Version 3 from 1500 to 2019.

Google provides a web-based service to search the frequencies of words by years on Google Books Ngram Viewer[1]. For every concept in the dataset, lemmas of the synset are listed by the NLTK WordNet library correspondingly. It can include all the words within the concept so that the frequency of a concept is more complete to represent its feature. Unfortunately, there is no official API for querying frequencies in a large-scale productive mode. The frequency data has to be obtained by a web crawl that posts requests for the frequency of a lemma and gets its response. The response can be parsed and analyzed to have valid frequency data. The returned data contains frequencies of a word(lemma) in the given period of years.

However, it is found that continuous requests to the Google Ngram Viewer would trigger an exception of *503 Service Unavailable* and respond to the null data. The reason is that Google set a limitation of request times to protect its server and services. The policy of the

[1] `https://books.google.com/ngrams`

Google server request limit is discovered to be likely 75 requests every 560 seconds. To solve this technical problem, we firstly implemented to set sleep time between every request, but it takes over 4 hours to query all the dataset. According to the policy of Google service, an optimal crawler is implemented to speed up the procedure of the querying. It reschedules the sleep strategy to 72 requests and then waits 580 seconds every round instead of sleeping 10 seconds between each request. With the new strategy, the time of the querying reduces to 3 hours for all the concepts. Besides, the optimal crawler is encapsulated as a Python class which can automatically query the frequencies given a concept and a period of years. Moreover, it provides an option that can aggregate the frequencies of a concept in a range for years into the maximum, minimum, mean, and standard deviation. The aggregations may help feature engineering to be discussed in Section 4.3.2.

3.5 English Semantic Feature Database

Mentioned in Section 2.1.4, BART will be fine-tuned with semantic features of words. A project of producing English semantic features[1] provides a database of 4436 concepts with their semantic features by Buchanan et al. (4). We use the English Semantic Feature Database to obtain the semantic features of concepts and build a training set for fine-tuning BART. The database is organized with word pairs $(concept, feature)$ which represent the close relation of their meanings and other statistics on semantics. For example, a word pair *(abandon, desert)* presents that concept *abandon* has the close meaning to *desert* which is a feature of concept *abandon*. Statistics of a pair include the frequency and the normalized occurrence of its feature in the database and the sample size of its concept. The part of speech is also labeled following each pair. They built the database by examining the answers to concepts obtained from crowdsourcing and processing their feature frequencies respectively. We only utilize the pairs in the English Semantic Feature Database to build a dataset of concepts and their semantic features.

The entire database has 69284 records of the word pairs with the part of speech (POS) as well as the statistics on features and frequencies. The features have already been 'translated' to lemmas (lemmatization) using Snowball stemmer (36) by Buchanan et al. The pairs of lemmas with with all the parts of speech are adopted in fine-tuning the BART. The detail of the English semantic feature database is summarized in Table 3.3. The #records column is the total features of the concepts before lemmatization.

[1] `https://wordnorms.com/`

POS	#Concepts	#Records	#Lemma Pairs
Noun	3125	51923	32051
Adjective	663	7511	3929
Verb	548	8772	6045
Other	100	1078	591
Total	4436	69284	42616

Table 3.3: Summary of the English Semantic Feature Database

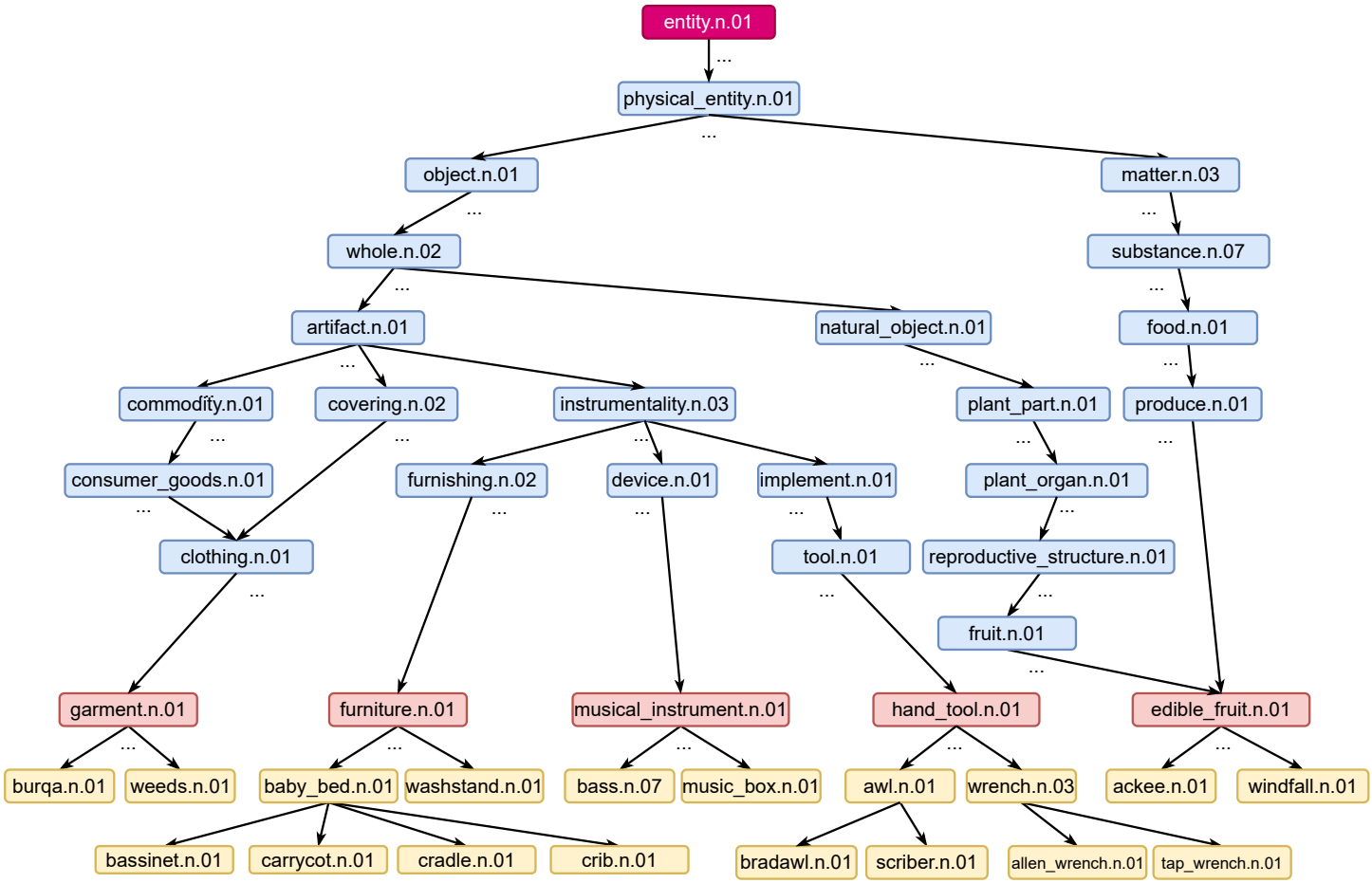

Figure 3.1: Hierarchy of concepts in WordNet (3)

4

Method

4.1 Classifier

Concepts are categorized into basic level or non-basic level. The classification task in this project is performed by a machine learning classifier, Random Forest with SMOTE algorithm. It is used to measure the performance of the synthetic features.

There are several reasons to adopt Random Forest with SMOTE algorithm. The classifier was developed and used in both Hollink et al. (26) and Henry (10). It has been proven that it is the best classifier for the basic level in Hollink et al. (26) and reused by Henry. Performance improved by the features in our work is easier to be compared with others using the same classifier. Another reason is the advantages of Random Forest itself. It introduces randomization that helps to avoid over-fitting. It can be trained fast and efficiently even with large-scale data. The input features can be both discrete and continuous variables without normalization. Moreover, after training and validation, it can return the importance of each feature which helps to analyze the effeteness of the features.

The Random Forest has 1400 Decision Trees as base learners trained with sub-dataset sampled with replacement from the dataset. Because of the method of bootstrap to build up the Random Forest, out-of-bag samples are feasible to estimate the generalization score of the classifier. For each Decision Tree in the Random Forest, Gini impurity is used to measure the quality of a split with a node. Gini impurity and entropy are equivalent in most cases. It is fast to calculate a linear function Gini impurity rather than entropy which contains logarithmic operations. The maximum depth of each tree is set to 50 which can control over-fitting and make the training fast. It requires that each split leads to at least two child nodes and each node has at least one instance from the training data.

We also use a Support Vector Machine(SVM) classifier as some semantic features are made up of vector-based embeddings. The Random Forest classifier was not good at classifying the word embeddings in our preliminary experiments. The SVM is trained with Radial Basis Function kernel, $\exp(-\gamma||x - x'||^2)$, where x and x' are both embedding vectors. After tuning by grid search, the best setting of the hyper-parameter γ is scaled by $\gamma = 1/(n * var)$, where n is the number of the vector dimension and var is the variance of the input matrix.

The SVM is only used for classification with the semantic feature of word embeddings in Section 4.4.1. The Random Forest as the benchmark of the method is the main classifier used to learn and predict the basic level.

4.2 Structural Feature Extraction

Structural features include lexical features of concepts and relational features of them extracted from WordNet. The synonymy, hypernymy, and hyponymy of a concept in WordNet convey semantic relations which reflect the senses of the concept with its superordinates and subordinates. According to the cognitive economy in Rosch et al. (5), the relational features would be important indicators to classify whether a concept is in the basic level that carries the most information and costs minimal effort. Moreover, as discussed in Section 2.1.1.1, the relational features in the hierarchy, WordNet, naturally represent a correlational structure of the real-world knowledge which is significant in the basic level theory.

The basic level can be learned also from the lexical features. As discovered by (37), one of the characteristics of the basic level concepts is that they are generally denoted by shorter and more polysemous words. Therefore, the character length of lemmas in a concept and the number of the lemma polysemies would be important features for predicting the basic level concepts.

Hollink et al. (26) and Henry (10) both considered these structural features. Their data of lexical features and relational features are referred to in our work. Only some of the WordNet features in Henry's work are selected and reused in our method. The following structural features are extracted and to be trained by the classifier. The number of the indirect hyponyms of a concept and the mean character length of lemmas in a concept are not reused because they did not improve the performance of predicting the basic level in our preliminary experiments. The sum of lemma frequencies in a concept is neither selected as the frequency feature is extracted and analyzed independently in Section 4.3.

- The number of the direct hypernyms of a concept
- The number of the total(direct and indirect) hyponyms of a concept
- The normalized number of part-whole relations related to a concept
- The normalized depth of a concept from the root synset
- The normalized character length in the gloss of a concept
- The shortest character length of lemmas in a concept
- The number of lemmas in a concept
- The maximal number of polysemies of lemmas in a concept

4.3 Concept Frequency

For the concept frequency feature, we focus on the sources of frequencies of concepts. To answer the first research question, corpora with different characteristics are firstly compared to extract the frequency features which contribute the most to the performance. Then, according to the most reliable corpus characteristics of predicting the basic level, frequencies of the concepts are acquired (in Section 3.4) and processed by feature engineering.

4.3.1 Corpus Characteristics Comparison

Frequencies of concepts can be extracted from various corpora. Roach et al. found that the basic level concepts are prominently used in daily communications, especially in communication with children. She also argued that the basic level could be the earliest categories sorted and named by children (5). Therefore, concept frequencies extracted from spoken discourses and child target audience corpora might improve the performance of basic level prediction. The corpus characteristics compared are the discourse type, target audience, and size of a corpus. The comparisons should take the size of the original corpus into account, as discussed in Section 3.3.

Unlike in Henry's work (10) which adopted a range of statistics to calculate frequencies, the frequency of a concept in this method is purely defined by occurrences of its lemmas. The sum of occurrences of the lemmas in a corpus of a given size represents the frequency of the concept. We did not choose the average of the occurrences as the averaged values

are extremely small which is harmful to the classification performance by Random Forest. The sum of occurrences in a corpus is the frequency feature of a concept directly fed to the classifier.

The first comparison concentrates on the size of a corpus. The hypothesis is that performance would become better when the size of the corpus increases. Based on the structural features, the frequency features are verified respectively by the benchmark Random Forest. With the different types and sizes of corpora sampled, multiple classifiers are trained and validated under the models and experiment settings designed in Section 5.1.

The second comparison focuses on the discourse type of a corpus. The hypothesis is that performance of the classifier trained by the frequency feature from a spoken corpus is better than that from a written corpus. This is inspired by the previous research that the basic level concepts are likely to be mentioned in daily communications and be the most used in languages (5). Intuitively, frequencies of the basic level in a spoken corpus would perform better and be more important than those in a written corpus. Under this assumption, the frequency feature from a spoken corpus is a more important and effective feature for the classifier. The comparison is conducted with a series of Wilcoxon rank-sum tests.

The third comparison concentrates on the target audience of a corpus. The experiments in Rosch et al. (5) showed that basic level concepts are the first used by children developing language. According to this statement, the hypothesis is that performance by the frequency feature from a corpus specific to children is better than that from a general audience corpus. Similar to the second comparison, the performance of the benchmark classifier trained with the frequency feature from a child-specific corpus and from a general audience corpus is compared by Wilcoxon rank-sum tests.

It is worth noting that both the second and the third comparisons consider the size of the frequency source, which is regarded as a primary corpus characteristic in this method. The design of the experiments is clarified in Section 5.2.

4.3.2 Frequency from Google Ngram

According to the results of the Wilcoxon rank-sum tests in Section 6.1.2, it would be better to use a large, written, and general audience corpus as the source of concept frequency. Google Book Ngram Corpus, is the largest corpus of printed publications available for public research. Therefore, the frequency features for predicting the basic level are extracted from Google Ngram. Same as the decision on the frequency source by Hollink et

al. (26) and Henry (10), they both selected Google Ngram to extract frequencies because the frequency feature from it was the strongest individual feature among their experiments.

The frequency of a concept is the sum of frequencies of its lemmas in Google Ngram. By the data acquisition in Section 3.4, the frequency of a concept by year can be returned with an array whose elements represent the frequencies of the concept each year in Google Ngram. To discover whether the time period affects the performance, frequencies from Google Ngram corpus in the recent (based on 2019) 1 year, 5, 10, 20, 50, 100, 200, 400, and 500 years are gathered and stored for feature engineering. The frequencies in each array can be aggregated into the mean and maximal values as the features. The mean frequency would represent the average occurrences of a concept during a certain range of years while the maximal one would show how much was the most significant used in those years. After the processing, two groups of the frequency features of a concept each with 9 values are respectively the mean frequencies and the maximal frequencies among the 9 time periods.

The classifier keeps the same corpus characteristics comparison in Section 4.3.1 except for the frequency feature selected. That is to say, the structural features remain as the base and pop each aggregated frequency feature into the training, shown in Figure 4.1. Hence, the performance of each Google Ngram frequency feature can be compared to discover whether there would be some patterns related to the time periods.

To find the best setting of the frequency features, we perform wrapper methods of the feature selection. Bottom-up and top-down approaches are deployed to check which combination would perform best with the metric of Cohen's kappa score. The importance of the features in the best setting from the benchmark Random Forest classifier is ranked and analyzed in Section 6. The best combination of the frequency features is added to the synthetic features together with the structural features and used to train the final classifier.

4.4 Generative Semantic Features

In our context "semantics" does not cover the general meaning of a concept. Because some of the structural features as described in Section 4.2 convey the general semantics from the lexical features or the relational features. "Semantic feature" in this context stands for the semantic representation of a concept which is key to models of semantic memory for facts (4) (38). In other words, semantic features in our work indicate the overlapping attributes of a concept defined by semantic similarity, which are regarded as cues in Section 2.1.1.2. For example, semantic features of a concept of *cat* might be *animal*, *pet*, *tail*, and *fur*.

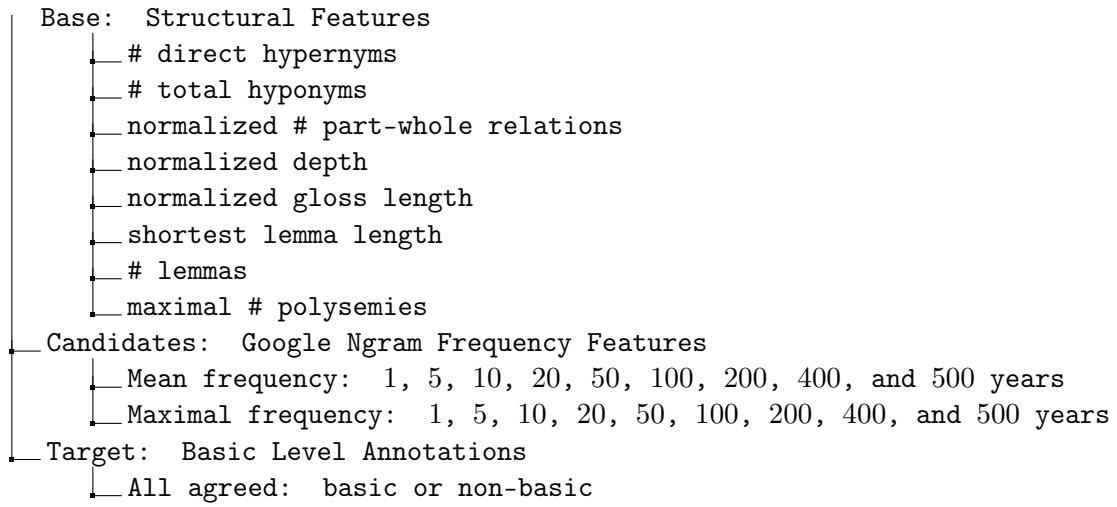

Figure 4.1: Frequency Feature Schema

These features convey the most common and regular descriptions of a *cat*. Moreover, the semantic features might cover shapes, appearance, uses, gender, locations, characteristics, etc. The aims of generating the semantic features are to measure the similarity of concepts and to create cues of concepts for predicting the basic level.

To learn the basic level from the semantic features, two methods are proposed to extract such semantics namely word embeddings and cues generated from BART. The rationales behind the two methods are: (1) Word embeddings, Word2vec, would provide effective semantic features for predicting the basic level because it can measure a latent semantic distance between concepts in a hierarchy. The semantic distance between a concept and its semantic features is close. For example, concept *cat* and feature *animal* or *tail* is close in semantic distance but is far from feature *electricity*. (2) Generating the semantic features from the fine-tuned BART as cues would improve the accuracy of the prediction by using cue validity. This is inspired by the finding by Rosch et al. (5) that basic-level categories have high cue validities and is based on the improved F1 score of detecting the basic level by BART in Chen and Teufel's (1).

4.4.1 Word Embeddings

Word embeddings are from *ConceptNet Numberbatch 19.08* trained by Word2vec, which is a task to represent words in the form of real-number vectors. The semantics of concepts is contained implicitly in the vectors. Intuitively, we use this model to compute the vector of a concept as looking up in a dictionary. Each lemma of concepts is converted to a 300-

dimension vector. Unfortunately, there are 63 concepts in multi-word grams which are not contained in the vocabulary of the pre-trained *ConceptNet Numberbatch 19.08* vectors. Originally, it is required to continue to train the model with sentences including these missing multi-word grams. However, 7.5% of the annotated concepts can not find entries, only one of which is the basic level. The missing concepts are eliminated from the dataset for convenience and the rest 776 concepts (174 of them are the basic level) are the training data.

Because the benchmark classifier used is Random Forest with SMOTE algorithm, it is not a good idea to feed the 300-dimension vectors into the model directly. The reason is that the vectors contain semantics implicitly, unlike the structural features and the frequency features explicitly show the attributes. There are two ways to represent such semantics: One is to adopt the vectors directly which means they could be learned by training a Support Vector Machine as a classifier. The second is to use feature engineering.

However, SVM is turned out from our experiments that the performance of classifying the basic level is not as good as Random Forest. The detailed results and interpretations are discussed in Section 6.3.1.1.

Feature engineering: alternatively semantics can be derived by measuring distances between semantic features. An aggregation method is proposed to extract the semantic features. Lemma distance is defined by the cosine similarity of vectors of two lemmas. Distance-based semantic features of a concept are calculated from lemmas in the concept and its hypernyms based on the lemma distance. The concept distances are then aggregated by mean, minimum, maximum, and standard deviation. For example, here is a hierarchy of three concepts in Figure 4.2 to calculate semantic features from their word embeddings. If semantic features of Concept *adjustable wrench* is required, cosine similarities as lemma distances between Lemma *wrench* and *adjustable wrench*, *wrench* and *adjustable spanner* as well as lemma distances between *spanner* and the other two lemmas are calculated. Then the mean, minimum, maximum, and standard deviation of the four lemma distances can become the distance-based semantic features.

The semantic features from word embeddings by Word2vec are the four concept distance aggregations of a concept. To find out whether distance-based semantic features improve the performance, we train the benchmark classifier Random Forest with SMOTE algorithm.

4.4.2 Generate From BART

Besides extracting semantic features from word embeddings, we suppose that concepts could be characterized by cues. The cues of a concept give properties, categories, at-

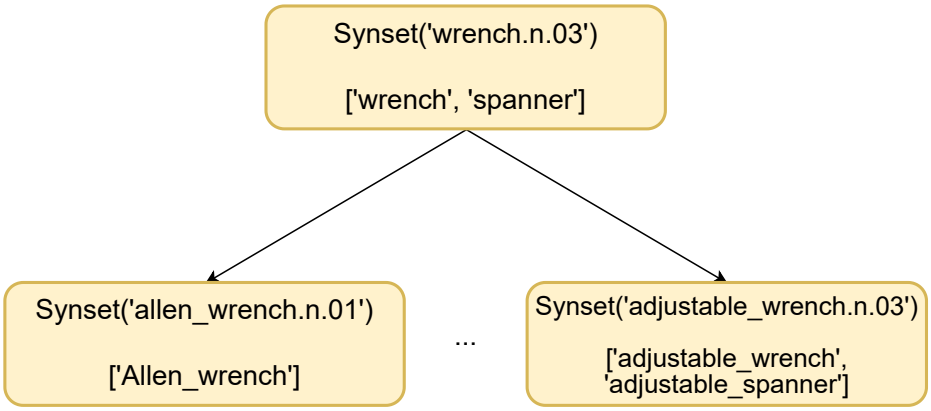

Figure 4.2: Example of calculating semantic features

tributes, and some characteristics to the concept. Inspired by Machine Translation, these textual semantic features can be generated by a sequence-to-sequence machine learning task. Traditional Machine Translation learns the mapping relations from a source language to a target language. It uses transformers to learn the context both in the source and the target. The autoencoder constructs a sequence-to-sequence architecture to learn the mapping relations. In our work, the source is concepts to be predicted and the target is a series of semantic features. The mapping relations are learned from the processed pairs in the English Semantic Feature Database. Therefore, generated target cues represent the semantic features of the source concept rather than its synonyms. One famous pre-trained model BART provides a good tokenization tool as well as a base model to fine-tune for semantic feature generation.

The training data for fine-tuning is from the English Semantic Feature Database (4). We only select concepts and their corresponding lemmatized features as *(concept, feature)* pairs to build up the fine-tuning dataset. To make the fine-tuning easier, we define a class of dataset inherited from Torch Dataset [1] to wrap the processed data into a set of a dictionary. Each dictionary would be a mapping from a word to its semantic features. The dataset class is also implemented with getLength and getItem functions.

After the processing and wrapping of the fine-tuning dataset. The generation is composed of three phases: tokenization, fine-tuning, and generating illustrated in Figure 4.3.

- **Tokenization**, as discussed in Section 2.1.4, we use Initialized Encoder to tokenize concepts. BART is here used as a tokenizer to obtain a token identification of each

[1] `https://pytorch.org/docs/stable/data.html#torch.utils.data.Dataset`

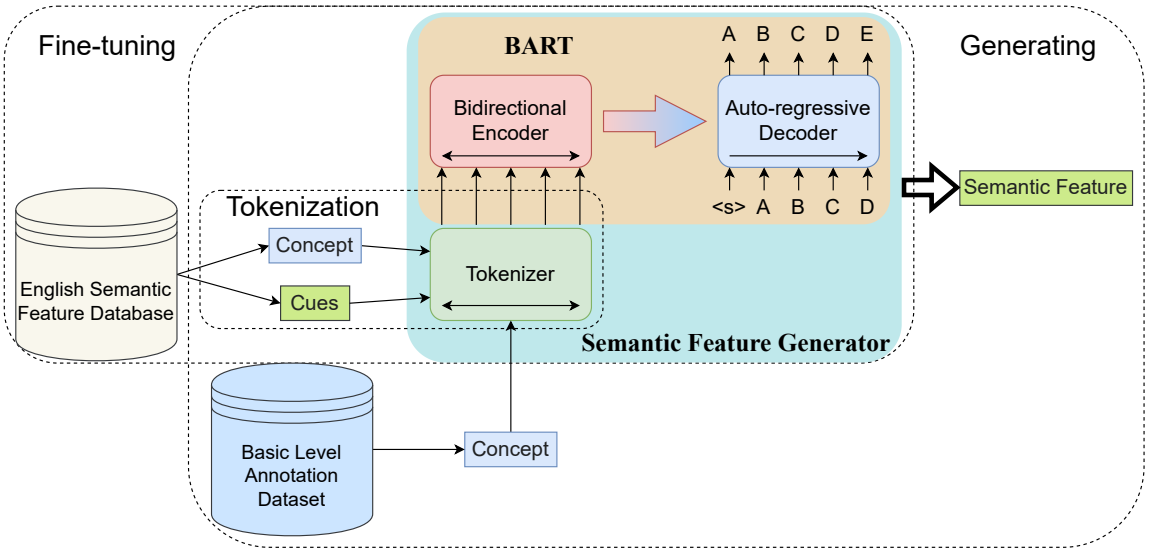

Figure 4.3: Semantic Feature Generation Pipeline

concept and its semantic features. The token identification is in form of a real-number tensor. It encodes text-based information into tensor-based numerical information which contains semantics pre-learned in BART.

- **Fine-tuning**, following the architecture of BART, we build a sequence-to-sequence trainer to learn the mapping in the English semantic feature database. The hyper-parameters are shown in Table 4.1. The metric is SacreBLEU which provides BLEU scores used to evaluate Machine Translation models (39). We aim to save the best two fine-tuned models during training. The two are used to translate lemmas into their semantic features.

- **Generating**, using one of the best fine-tuned models, we can generate semantic features of the concepts in the annotation dataset. The concepts in the dataset can be fully fitted in this pipeline for semantic feature generation.

Although we are able to generate semantic features, they are text-based which are not easy to learn directly. It is more reasonable to transform the textual feature into a numerical one that could utilize the semantics. I come up with an indicator that reflects the original basic level theory, namely cue validity. According to Rosch's statement, cue validity can be a probabilistic indicator which is the validity of a given cue as a predictor of a given

Hyper-parameter	Value
evaluation strategy	*after epoch*
learning rate	$2e-5$
train batch size	8
evaluate batch size	8
weight decay	0.01
checkpoint number	2
train epochs number	3
predict with generate	*true*

Table 4.1: Hyper-parameter Setting for Fine-tuning BART

category, the basic-level category (5). To make it easy to understand, cue validity is from conditional probability to indicate how likely it would be at the basic level.

As discussed in Section 2.1.1, there are different formulations of the cue validities. To have a measurement for the cue validity in the project, based on the formal probabilistic conception, here is a formula for computing cue validity given the cue and knowing whether a concept is the basic level:

$$val(cue) = P(BL|cue) = \frac{P(BL \wedge cue)}{P(cue)} \tag{4.1}$$

Since a concept does not have the only cue in most cases, the cue validity of one concept, $CV(concept)$, is defined by a sum of the cue validities of a group of cues, which are the attributes of lemmas of the concept, with Equation 4.1:

$$CV(concept) = \sum_{l \in lemmas(concept)} \sum_{cue \in attribute(l)} \frac{P(BL \wedge cue)}{P(cue)} \tag{4.2}$$

The cue validity of a concept is no longer a probability but an accumulation of probabilities. The same hierarchy can be taken as an example of defining basic level by cue validity of concepts, in Figure 4.4. Here are the lemma attributes and a cue validity of each concept in the hierarchy. The cues shown in the figure are mostly from a database by (4) or generated by Section 4.4.2 if concepts are not in the database. It reveals that the concept of cat and the concept of dog show the greatest two cue validities which indicate they are the basic level. The result keeps the same as that in the approach of cognitive economy.

The number of the cues and the cue validity can be the semantic features of a concept according to the method. These two features are input to train the benchmark Random Forest classifier for predicting the basic level.

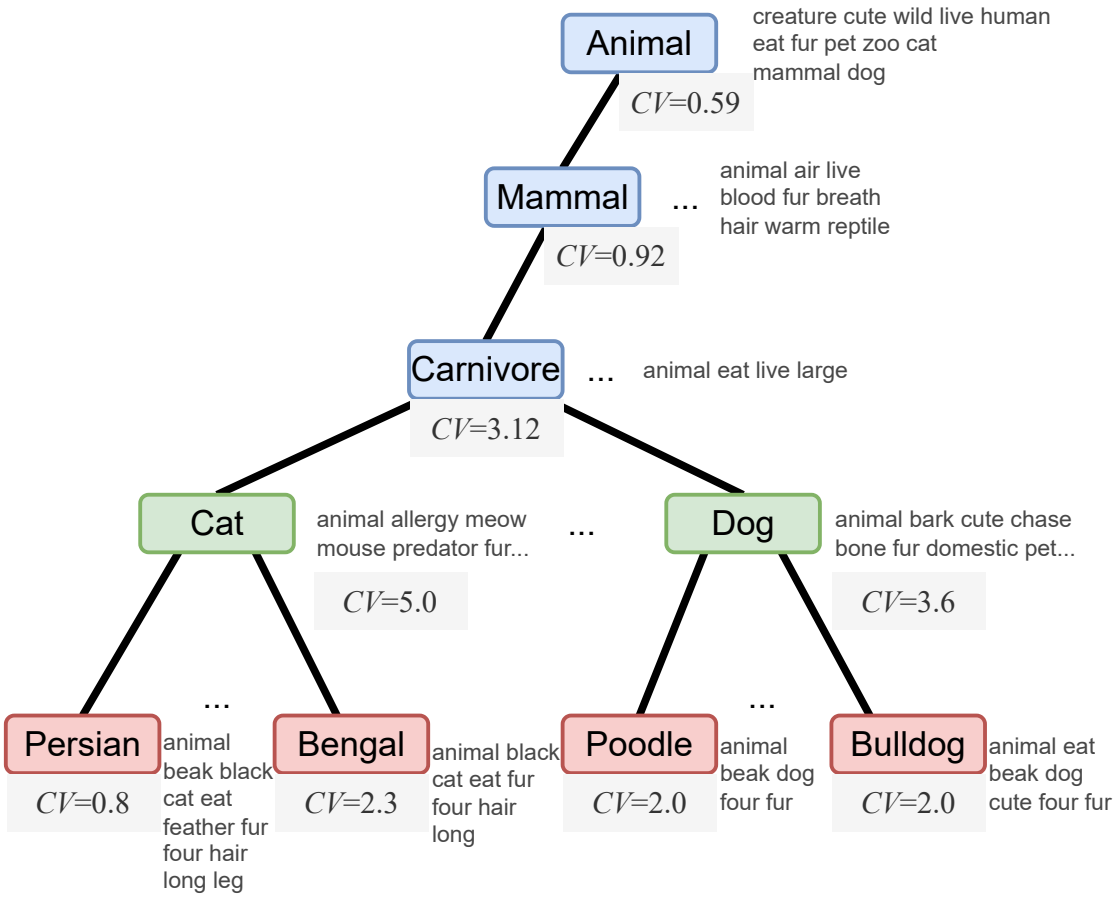

Figure 4.4: Concept hierarchy with cue validities example(4)

5

Experiment Setting

5.1 Dataset and Model Setup

The basic level annotations and the synthetics features of the concepts constitute the final dataset to be used to train the Random Forest classifier. Three types of models are performed to test improvement of predicting basic level from the synthetic features. In this section, metrics measuring the performance of models, splitting of the dataset, and the three models will be discussed.

Cohen's kappa score(40) and balanced accuracy score(41) are used as metrics for evaluating the performance of the models with imbalanced data. Cohen's kappa measures the inter-rater reliability for the basic level or not. Specifically, it indicates how well the model predicts the basic level correctly compared to predicting randomly by chance. Balanced accuracy is useful to evaluate how good a binary classifier is when trained with imbalanced data. It considers sensitivity, which is the true positive rate, and specificity, which is the true negative rate.

For each experiment, 10-fold cross-validation is used to train and evaluate the model. Due to the imbalance of the annotation data, Stratified K-fold[1], a variation of K-fold cross-validation which samples each set to contain the same percentage of the basic level as the whole dataset, is implemented to split the dataset into training data and validation data with 10 groups. Under this setting, every experiment will return 10 groups of Cohen's kappa scores and balanced accuracy scores. The two averaged values of the scores respectively are used to evaluate the performance of the model.

The experiments include the effectiveness of each type in the synthetic features based on the structural one, Table 5.1. To verify the method and the synthetic features within and

[1] `https://scikit-learn.org/stable/modules/cross_validation.html#stratified-k-fold`

	Feature type
EX_F_1	Structural features
EX_F_2	Structural features + Frequency features from Google Ngram
EX_F_3	Structural features + Semantic features by word embeddings
EX_F_4	Structural features + Semantic features generated by BART

Table 5.1: Experiment Settings for Feature Effectiveness

	Training Data	Testing Data
GlobalModel	All 5 domains	All 5 domains
LocalModel	*hand tool*	*hand tool*
	edible fruit	*edible fruit*
	musical instrument	*musical instrument*
	furniture	*furniture*
	garment	*garment*
TransferModel	*edible fruit, musical instrument, furniture, garment*	*hand tool*
	hand tool, musical instrument, furniture, garment	*edible fruit*
	hand tool, edible fruit, furniture, garment	*musical instrument*
	hand tool, edible fruit, musical instrument, garment	*furniture*
	hand tool, edible fruit, musical instrument, furniture	*garment*

Table 5.2: Model Settings

across domains, three models are set up for evaluating and comparing the performance. They are named GlobalModel, LocalModel, and TransferModel, described in Table 5.2.

5.1.1 GlobalModel

GlobalModel is trained and tested with the data from all the five domains in Table 3.1. This model can fully use the annotated concept we have. In other words, it can have as much data as possible to participate in the training. By GlobalModel, the overall performance of predicting the basic level in a hierarchy will be revealed. The results would indicate whether it is effective to add the synthetic features in training and how much it improves or hurts the accuracy for all the domains.

5.1.2 LocalModel

LocalModel is trained and tested the classifier with the concepts in the same domain. Therefore, there will be five models trained within each LocalModel setting. The result of each LocalModel will indicate whether it is effective to train with the synthetic features within a specific domain. The results by the five domains can then be averaged only to show the influence introduced by the different kinds of features on the five domains. Feature importance in every Random Forest classifier is returned for comparing contributions of the synthetic features among the different domains.

5.1.3 TransferModel

TransferModel is trained with concepts within four of the five domains and tested on the rest. Similar to LocalModel, there will be five TransferModel trained during an experiment. However, it does not need to set up the 10-fold cross-validation because training data and validation data have been split by the definition. The result of each experiment is the averaged metrics from TransferModels of the five domains. TransferModel aims to verify the generalization of the method. The accuracy on unseen domains means whether the trained model is appropriate to predict the basic level under other domains of knowledge. It can help to detect the basic level in a large-scaled hierarchy, all concepts in WordNet.

5.2 Wilcoxon Rank-Sum Test

To answer the first research question about the relation between prediction performance and the size of corpora, it requires finding whether there is a dependency between the corpus size and the metrics, Cohen's kappa, or balanced accuracy. Wilcoxon rank-sum test, also known as Mann-Whitney U test(42), is performed to test the null hypothesis that the prediction performance of Cohen's kappa values and balanced accuracy by different sizes of the same corpus are from the same distribution.

The experiment focuses on the source of frequency features. Before conducting Wilcoxon rank-sum test, the corpora are sampled into different sizes in Table 5.3 and used to calculate frequencies. Each corpus in the scheme is sampled 50 times. For example, BNC is sampled into the word counts of 1 million, marked *BNC 1M*, 50 times. Therefore, there are 500 sampled corpora in total. The sampled corpora will be the sources of the frequency features which are used to train the Random Forest classifier.

Corpus	1M	2.4M	5.7M	100M
BNC	√	√	√	√
CHILDES	√	√	√	
CABNC	√	√		
KBNC	√			

Table 5.3: Corpora Sampling in Different Sizes

After sampling, the classifier is trained and tested by the structural features and the frequency feature. With the model setting in Section 5.1, frequency features from different sampled corpora can be used to train and test with the three models. Each corpus in a specific size leads to 50 results with each model. Totally, there will be 1500 groups of Cohen's kappa and balanced accuracy results from the corresponding models and frequency sources.

Wilcoxon rank-sum test is carried out to test two aspects of hypotheses about the size and the type of corpora. The experiment settings are described in Table 5.4 and Table 5.5. The initial setting EX_W_0 is to have the Cohen's kappa and the balanced accuracy of each model from the samples in Table 5.3. The first setting EX_W_1 is to compare the results from the same corpus but with different sizes. The second setting EX_W_2 is to compare the results from the same discourse type of a corpus and different sizes. The third setting EX_W_3 is to compare the results from the same target audience of a corpus and different sizes. The last setting EX_W_4 is to compare the results from the same size but different discourse types and target audiences of corpora.

5.3 Experiments on Synthetic Features

As described in Table 5.1, There are four settings of the experiments checking whether and how much the synthetic features would improve the performance of predicting the basic level. Each of the settings check the features with the three models in Table 5.2.

5.3.1 Structural Features

Experiment EX_F_1 only trains the Random Forest classifier with the structural features in Section 4.2. It is used as the baseline compared with the performance by frequency features from Google Ngram and the semantic features in terms of improvement and growth rate.

	Corpus	Size
EX_W_0	KBNC	1M
	CABNC	1M, 2.4M
	CHILDES	1M, 2.4M, 5.7M
	BNC	1M, 2.4M, 5.7M, 100M
EX_W_1	CABNC	1M - 2.4M
	CHILDES	1M - 2.4M
		1M - 5.7M
		2.4M - 5.7M
	BNC	1M - 2.4M
		1M - 5.7M
		1M - 100M
		2.4M - 5.7M
		2.4M - 100M
		5.7M - 100M
EX_W_2	Written	1M - 2.4M
		1M - 5.7M
		1M - 100M
		2.4M - 5.7M
		2.4M - 100M
		5.7M - 100M
	Spoken	1M - 2.4M
		1M - 5.7M
		2.4M - 5.7M
EX_W_3	General	1M - 2.4M
		1M - 5.7M
		1M - 100M
		2.4M - 5.7M
		2.4M - 100M
		5.7M - 100M
	Children	1M - 2.4M
		1M - 5.7M
		2.4M - 5.7M

Table 5.4: Experiment Settings for Wilcoxon Rank-Sum Test

	Size	Discourse Type / Target Audience
EX_W_4	1M	Written - Spoken
		General - Children
	2.4M	Written - Spoken
		General - Children
	5.7M	Written - Spoken
		General - Children

Table 5.5: Experiment Settings for Wilcoxon Rank-Sum Test

5.3.2 Frequency Features

Experiment EX_F_2 trains the Random Forest classifier with the structural features and the frequency features from Google Ngram. As discussed in Section 4.3.2, we first find and select the best combination of the frequency features in the three models respectively. Then, the classifiers with the three model settings are trained using the best selections of the frequency features. The results of each model are compared with the baseline to check how much the frequency features from Google Ngram improve the performance.

5.3.3 Semantic Features

5.3.3.1 Word Embeddings

Experiment EX_F_3 has two sub-experiments: vector-based and distance-based. The EX_F_3:vector-based trains the SVMs in the three models with the structural features and the 300-dimension vectors which implicitly stand for the semantic feature of concepts. The results are compared with the baseline to see whether the original word embeddings by Word2vec could improve the performance. The EX_F_3:distance-based train the Random Forest classifiers with the structural features and the aggregations of concept distances which explicitly convey the semantic feature of concepts. The results are similarly compared with the baseline to check how much the aggregated features from the word embeddings improve the performance.

5.3.3.2 Generated by BART

Experiment EX_F_4 consists of three steps. The first step is fine-tuning BART and generating cues of the concepts in the basic level annotation dataset. The second step is calculating the cue validity of every concept. The last step is training the Random Forest classifiers in the three models. We only discuss the results of predicting the basic level from

the last step[1]. The results are compared to the baseline to check how much the semantic features of generated cues and the cue validity improve the performance of predicting the basic level.

5.4 Detect in WordNet

To predict the basic level under a large-scale data environment for the uses in applications, we design EX_WN to detect the basic level concepts from all the synonyms in WordNet under the branch of *entity.n.01* which are the entire concepts with the root of *entity.n.01* shown in 3.1. The model with the synthetic features resulting in the highest Cohen's kappa is used to detect the basic level in WordNet. The detection results are made into a dataset of the basic level concepts. The dataset will be evaluated and compared with Henry's (10) and Hollink et al's (26) in Section 6.

[1]The results of the former two steps can be found in the GitHub.

6

Results & Evaluation

6.1 Tests on Corpus Characteristics

6.1.1 EX_F_1: Structural Features

In order to set up a baseline for our work, we first tried to reproduce the work Henry's work (10). The performance of the experiment settings is summarized in Table 8.1, Table 8.2, and Table 8.3. We regard the results as the baseline for predicting the basic level with different features from the methods proposed.

The results are not exactly the same as Henry's because we only chose the WordNet features, the structural features in this thesis, to train the classifier. The trends of Cohen's kappa and balanced accuracy are similar across the experiments. They are used to verify whether the new frequency and semantic features have positive effects to improve the performance on predicting the basic level.

6.1.2 Wilcoxon Rank-Sum Test

Because the null hypothesis H_0 is tested across several results of the experiment settings, according to (43), we have to apply Bonferroni correction to the significance level of $\alpha = 0.05$ with the number of the null hypotheses m. The correct significance level is:

$$\alpha' = \frac{\alpha}{m} \tag{6.1}$$

As shown in Table 5.4, there are 10 H_0s in EX_W_1, so the significance level is $\alpha' = 0.005$. Similarly, $m = 9$ in EX_W_2 and EX_W_3, α' is both 0.006. In Table 5.5, the results from different corpus samples are tested for 3 times. Therefore, α' is 0.017 in EX_W_4.

Corpus	1M	2.4M	5.7M	100M
BNC	0.680	0.675	0.675	0.689
CHILDES	0.666	0.674	0.675	
CABNC	0.674	0.672		
KBNC	0.686			

Table 6.1: Averaged Cohen's Kappa of EX_W_0-GlobalModel

Corpus	1M	2.4M	5.7M	100M
BNC	0.650	0.657	0.655	0.662
CHILDES	0.643	0.647	0.649	
CABNC	0.643	0.645		
KBNC	0.633			

Table 6.2: Averaged Cohen's Kappa of EX_W_0-LocalModel

6.1.2.1 EX_W_0: Performance with Samples

After sampling the four corpora 50 times with different sizes as discussed in Section 5.2, each sample is the source of frequency features that are used to train the classifier together with the structural features as the base. The overall results of the three models are shown by box plots in Appendix B.1. The data in the box plots of the GlobalModel is directly the results of the two metrics testing on the five domains, while the data in the LocalModel and the TransferModel is the average results tested on the five settings shown in Table 5.2. The Cohen's kappa of different models are shown in Table 6.1, Table 6.2, and Table 6.3, their balanced accuracy are shown in Appendix B.1. The averaged performance metrics of each frequency source in the three model settings are summarized respectively in Figure 8.1 Figure 8.3, and Figure 8.5.

Corpus	1M	2.4M	5.7M	100M
BNC	0.532	0.539	0.543	0.556
CHILDES	0.524	0.521	0.518	
CABNC	0.510	0.516		
KBNC	0.531			

Table 6.3: Averaged Cohen's Kappa of EX_W_0-TransferModel

The null hypothesis is that Cohen's kappa value would get greater with an increment of the size of the corpus. Among the three models, LocalModel on the left of Figure 8.3 shows the best-averaged kappa. However, it might be not useful to predict concepts with unknown domains. For GlobalModel on the left of Figure 8.1, KBNC the samples in 1 million show good results compared to the other corpora in a relatively small size. Except for that, among all the other settings, BNC shows the best performance, especially with BNC 100M. For TransferModel on the left of Figure 8.5, CHILDES even has a continuous descending trend.

The performance shown in Figure 8.2, Figure 8.4, and Figure 8.6 depict distributions of the performance of the trained modules with different settings of the samples. For the LocalModel, BNC performs the best in terms of the median of Cohen's kappa. With the increment of the size, kappa values tend to have lower variances. BNC and CABNC perform better than the other two children specific corpora. But it is the only case in the LocalModel. For GlobalModel, BNC and KBNC perform similarly well. It is an exciting finding that the written corpora would perform better than the spoken ones in the GlobalModel also in the TransferModel shown in the figures. For TransferModel, BNC is the best frequency source among them again.

6.1.2.2 EX_W_1: Same Corpus Different Sizes

In order to verify the existence of deep relationships in the three models individually. By just looking at the plots in Figure 8.1, Figure 8.3, and Figure 8.5, it is difficult to draw any conclusion, we have thus decide to use statistical method to test the hypotheses. To make it convincing, we use a statistical method to test hypotheses. As described in Section 5.2, we performed Wilcoxon rank-sum tests on the testing results by the classifiers trained with frequency features from corpora in different sizes.

The first group of results is about the performance with the same corpus but with different sizes. The null hypothesis H_0 is that Cohen's kappas under the same model but with different two sizes of a given corpus are from the same distribution. The alternative hypothesis H_1 is that the kappa values with a smaller corpus size are less than those with a larger size.

The p-values of the Wilcoxon rank-sum tests on Cohen's kappa from EX_W_1-GlobalModel are in Table 6.4. The results are not significant enough to reject H_0. We can not say the performance becomes better with a larger corpus in the GlobalModel. The situation is the same in the LocalModel in Table 6.5. In the TransferModel, Table 6.6, there are more

Corpus Size	CABNC	CHILDES	BNC
1M-2.4M	0.822	0.000	0.993
1M-5.7M		0.000	0.983
1M-100M			0.000
2.4M-5.7M		0.342	0.590
2.4M-100M			0.000
5.7M-100M			0.000

Table 6.4: P-values of EX_W_1-GlobalModel on Cohen's Kappa

Corpus Size	CABNC	CHILDES	BNC
1M-2.4M	0.122	0.036	0.001
1M-5.7M		0.001	0.007
1M-100M			0.000
2.4M-5.7M		0.075	0.758
2.4M-100M			0.003
5.7M-100M			0.000

Table 6.5: P-values of EX_W_1-LocalModel on Cohen's Kappa

p-values lower than $\alpha' = 0.005$ accepting H_1 that a larger size corpus as the frequency source leads to a better performance of predicting the basic level.

6.1.2.3 EX_W_2: Same Discourse Type Different Sizes

Similar to EX_W_1, Wilcoxon rank-sum tests are performed to test the null hypothesis H_0 that Cohen's kappas under the same model, the same discourse type, but different two sizes of a given corpus are from the same distribution. The H_1 is that a corpus with the larger size outcomes the better Cohen's kappa. The significance level is $\alpha' = 0.006$

The results of the tests with Cohen's kappa for GlobalModel, LocalModel, and Transfer-Model are respectively shown in Table 6.7, Table 6.8, and Table 6.9. In the GlobalModel, there is only one test using the spoken corpora and three tests using the written corpora that reject H_0. Especially, only the written corpus with the sample size of 100 million, which is right *BNC 100M*, can reject H_0 as it performs the best over all the settings. Within the spoken corpora, 5.7 million samples also stand out which can be interpreted to gain from a larger size. In the LocalModel, most of the tests with the written corpora reject H_0 except the size of 2.4 million versus 5.7 million. The tests from the spoken corpora

Corpus Size	CABNC	CHILDES	BNC
1M-2.4M	0.000	0.959	0.005
1M-5.7M		1.000	0.000
1M-100M			0.000
2.4M-5.7M		0.962	0.084
2.4M-100M			0.000
5.7M-100M			0.000

Table 6.6: P-values of EX_W_1-TransferModel on Cohen's Kappa

Corpus Size	Written	Spoken	General	Children
1M-2.4M	1.000	0.016	0.993	0.937
1M-5.7M	1.000	0.003	0.863	0.865
1M-100M	0.000		0.000	
2.4M-5.7M	0.590	0.074	0.358	0.342
2.4M-100M	0.000		0.000	
5.7M-100M	0.000		0.000	

Table 6.7: P-values of EX_W_2 and EX_W_3-GlobalModel on Cohen's Kappa

indicate the same rejections as in the GlobalModel. In the TransferModel, no tests with the spoken corpora can reject H_0 but five of the tests with the written corpora reject H_0.

It is more meaningful to take the test results of TransferModel as references for predicting concepts than the other models. The reason is that in most cases prediction is done with concepts from different or unseen domains. The TransferModel experiments we designed and performed can simulate such a situation well.

6.1.2.4 EX_W_3: Same Target Audience Different Sizes

The null hypothesis H_0 that Cohen's kappas under the same model, the same target audience, with different two sizes of given corpora are from the same distribution. The alternative hypothesis H_1 is that the larger size leads to better performance, like in EX_W_2.

The results of the tests are outlined in Table 6.7, Table 6.8, and Table 6.9. In the GlobalModel, the frequency features from the general corpora perform similarly to them from the written corpora. This is mostly because BNC contributes the best performance and much exceeds the other corpora. Especially in the LocalModel and the TransferModel,

Corpus Size	Written	Spoken	General	Children
1M-2.4M	0.000	0.013	0.004	0.000
1M-5.7M	0.000	0.000	0.000	0.000
1M-100M	0.000		0.000	
2.4M-5.7M	0.748	0.074	0.003	0.075
2.4M-100M	0.003		0.000	
5.7M-100M	0.000		0.000	

Table 6.8: P-values of EX_W_2 and EX_W_3-LocalModel on Cohen's Kappa

Corpus Size	Written	Spoken	General	Children
1M-2.4M	0.000	0.129	0.001	1.000
1M-5.7M	0.000	0.246	0.000	1.000
1M-100M	0.000		0.000	
2.4M-5.7M	0.084	0.510	0.000	0.962
2.4M-100M	0.000		0.000	
5.7M-100M	0.000		0.000	

Table 6.9: P-values of EX_W_2 and EX_W_3-TransferModel on Cohen's Kappa

all the tests with the written corpora reject H_0. The children corpora do not have a consistent conclusion on whether to reject the null hypotheses.

From the results in EX_W_2 and EX_W_3, the written group and the general group reject the null hypothesis significantly. We can say that using a larger size general written corpus to predict the basic level is better than a smaller one for the TransferModel confidently. The situations are similar under the LocalModel and the GlobalModel. However, when looking at the children-specific corpora, the p-values are dramatically high. There is only a written children corpus KBNC with 1M records in the dataset. Moreover, frequency features from KBNC show pretty good performance in the Random Forest classifier as in Table 6.1 and Figure 8.1. This leads to a dilemma when testing the size within children-specific corpora.

6.1.2.5 EX_W_4: Same Size Different Types and Targets

EX_W_4 is performed to compare the performance of predicting the basic level by different discourse types and target audiences. The null hypothesis H_0 is that Cohen's kappas under the same model, the same size of corpora, but different types and targets are from

Corpus Characteristics	1M	2.4M	5.7M
Written - Spoken	0.000	0.103	0.566
General - Children	0.558	0.755	0.566

Table 6.10: P-values of EX_W_4-GlobalModel on Cohen's Kappa

Corpus Characteristics	1M	2.4M	5.7M
Written - Spoken	0.973	0.000	0.001
General - Children	0.000	0.008	0.001

Table 6.11: P-values of EX_W_4-LocalModel on Cohen's Kappa

the same distribution. The alternative hypothesis H_1 is the performance of the former corpus is greater than the performance of the latter. The significance level is $\alpha' = 0.017$.

The results of the tests are shown in Table 6.10, Table 6.11, and Table 6.12. In the GlobalModel, only the test between written corpora and spoken corpora with the same size of 1 million reject the null hypothesis. It is the opposite in the LocalModel that the other tests all reject H_0. In the TransferModel, all the p-values of the tests for written and spoken corpora are 0. We can confidently reject the null hypothesis and conclude that frequency features from written corpora perform better than those from spoken ones. Almost similar to Written - Spoken, the tests for general audience and children audience corpora with samples of 2.4 million and 5.7 million indicate that we can reject the null hypothesis and conclude that using general versions is better than specific to children ones. However, there is again an abnormal result with 1 million samples from the children-specific corpora, which implies that children-specific corpora perform better than general corpora under a size of 1 million.

Using the best performing corpus characteristics which are large, written, and for general audiences, we can see the improvement from Table 6.13 compared to the baseline. *BNC 100M* with the best corpus characteristics is used as the source of the frequency feature.

Corpus Characteristics	1M	2.4M	5.7M
Written - Spoken	0.000	0.000	0.000
General - Children	1.000	0.029	0.000

Table 6.12: P-values of EX_W_4-TransferModel on Cohen's Kappa

Model	With BNC 100M	Baseline	Improvement	Growth Rate
GlobalModel	0.690	0.673	0.017	+2.53%
LocalModel	0.662	0.640	0.022	+3.44%
TransferModel	0.556	0.521	0.035	+6.72%

Table 6.13: Best Corpus Characteristics Setting Performance on Cohen's Kappa

6.2 Frequency Features: EX_F_2

There can be multiple frequency features of concepts from Google Ngram discussed in Section 4.3.1. The performance of the frequency features is evaluated with the three model settings in Table 5.2.

The results of the test of each module are plotted respectively in Appendix C. In the GlobalModel (Figure 8.7), maximal frequency mainly behaves better kappa except for *1 year*. It is expected to select *100 year maximum* as the frequency feature. An unexpected phenomenon is that *5 year* and *100 year* have a different trend. Moreover, *100 year maximum* has the best score in the same time the *100 year mean* has the worst. One possible cause of this phenomenon is that the value of the maximal frequency of *100 year* is relatively much larger than the mean frequency and then it would have a dominant effect.

With the LocalModel (Figure 8.8), the mean frequencies are likely to perform better than maximal frequencies, which is different from GlobalModel results. The peak score is reached for *20 year mean*. *100 year mean* also performs well and has the second-highest kappa value. In the TransferModel (Figure 8.9), the trends of the mean frequencies and the maximal frequencies are interweaved. *20 year mean* has the best score then followed by two similar *100 year mean* and *100 year maximum*. However, the average value of kappas is lower than the GlobalModel's and the LocalModel's.

To discover the best frequency feature setting, a wrapper method of the feature selection is performed. We deploy bottom-up and top-down approaches to check which setting would perform the highest kappa value for each model. The selection results and importance of the features are shown in Appendix C.

After selecting the best group of the frequency features from Google Ngram, they are used to train the classifier with the three models. The results and improvement compared to the baseline (not using frequency features from Google Ngram) are shown in Table 6.14. The growth rates are calculated based on the baseline in EX_F_1. Compared with the

Model	With Google Ngram	Baseline	Improvement	Growth Rate
GlobalModel	0.714	0.673	0.041	+6.16%
LocalModel	0.712	0.640	0.072	+11.09%
TransferModel	0.590	0.521	0.069	+13.21%

Table 6.14: Results of EX_F_2 on Cohen's Kappa

Model	With Embedding Vector	Baseline	Improvement	Growth Rate
GlobalModel	0.551	0.673	−0.122	−18.13%
LocalModel	0.561	0.640	−0.079	−12.34%
TransferModel	0.407	0.521	−0.114	−21.88%

Table 6.15: Results of EX_F_3 with Vector-Based Features on Cohen's Kappa

improvement by the best performing corpus characteristics in Table 6.13, Google Ngram improves more because of the larger corpus size.

6.3 Semantic Features: EX_F_3, EX_F_4

6.3.1 EX_F_3: Word Embeddings

We use two methods to extract the semantic features from word embeddings and evaluate them using two different classifiers as described in Section 4.4.1.

6.3.1.1 Vector-Based Features

The first method to extract the semantic features is a 300-dimension vector of a concept. The vectors are trained using SVM. Results and improvement are shown in Table 6.15.

Using SVM as the classifier, the results from vector-based features do not perform better as expected, which confirms the results of Chen's (1). The embedding features trained by SVM even decrease the performance compared to the baseline trained by Random Forest without the embeddings. One possible reason is that SVM tends to classify the concepts with the similarity of lemmas. In other words, the concepts in the subordinate categories are more likely to share a higher similarity and form the support vectors. This potentially harms the harm to the binary classification task of the basic level.

Model	With Semantic Distance	Baseline	Improvement	Growth Rate
GlobalModel	0.713	0.673	0.040	+6.01%
LocalModel	0.648	0.640	0.008	+1.23%
TransferModel	0.531	0.521	0.010	+2.06%

Table 6.16: Results of EX_F_3 with Distance-Based Features on Cohen's Kappa

6.3.1.2 Distance-Based Features

The other extraction method is the aggregation of concept distances. The aggregated distances used as the semantic features are trained using Random Forest. The results and improvement are shown in Table 6.16.

All the three model tests show better Cohen's kappa values. Compared to only learning from the structural features, the distance-based semantic features from word embeddings help to obtain better performance. It proves that distance-based features can enhance the performance of predicting the basic level. The GlobalModel can learn more from the distance-based semantic features. However, the improvement from the distance-based semantic features for the LocalModel and the TransferModel is limited just by 1% and 2% increased.

6.3.2 EX_F_4: Generated by BART

The improvement from the generated semantic features is huge in the three models as seen in Table 6.17. The results evaluate the effectiveness of the fine-tuned BART generator and the formula of cue validity for predicting the basic level. Performance enhances the most in the LocalModel by 40.31%. This is due to a similar trend of the cue validities of the basic level concepts within the same domain. It is easier and more efficient to learn the trend of the cue validity than cross domains. That is why the improvement in the GlobalModel 30.91% and in the TransferModel 35.32% is not as much as in the LocalModel.

Cohen's kappa in the GlobalModel is 0.881 and in the LocalModel is 0.898, which both illustrate the best classification with Cohen's kappa of 0.740 among the related work discussed in Section 2.2. The performance in the TransferModel though is in the last place among the three models, it has been better than the best one in the baseline.

Model	With Generated Cues	Baseline	Improvement	Growth Rate
GlobalModel	0.881	0.673	0.208	+30.91%
LocalModel	0.898	0.640	0.258	+40.31%
TransferModel	0.705	0.521	0.184	+35.32%

Table 6.17: Results of EX_F_4 with Generated Cues by BART on Cohen's Kappa

6.4 EX_WN: Large-Scale Detection

The best performance of the synthetic features and the basic level annotation in the dataset are used to train the final classifier to detect the basic level in WordNet. The synthetic features include the eight structural features in Section 4.2, frequency features from Google Ngram in the time period of 50 years and 100 years, and semantic features from generated cues and the cue validity. Under *entity.n.01* in WordNet, 74,374 synsets are labelled with the basic level or the non-basic level.

From the prediction results, there are 9,085 synsets predicted as the basic level by the classifier. Approximate 12.2% of the synsets belong to the basic level. This proportion is less than that in the basic level annotation dataset, which is 20.8% of the concepts annotated with the basic level by agreement of the three annotators. Two similar datasets of the basic level created by automatic prediction are from Hollink et al. (26) and Henry (14). Both of them predicted the basic level under *entity.n.01* in WordNet. For validation, we can compare our dataset with theirs. Hollink et al.'s consists of 9,819 basic level concepts. The amount is close to our detection. We have 7,872 basic level concepts in common taking up 86.6% of our dataset. Henry's predicted basic level dataset has 15,401 basic level concepts, which is almost twice larger than ours. There are 8,588 shared basic level concepts between ours and Henry's, taking up 94.5% of our predictions. Henry's agrees on more basic level concepts in our dataset than Hollink et al.'s as a larger amount of basic level concepts in Henry's dataset.

7

Discussion

7.1 Comparing Corpus Characteristics

7.1.1 Discourse Type, Target Audience, and Size

SQ1: To what extent the discourse type and target audience of a corpus considering its size would affect the performance of predicting the basic level?

From the results of the experiments in Section 6.1, corpus characteristics do affect the performance of the prediction, which implies corpora with different characteristics can lead to different accuracy. However, the differences are not significant among the discourse type of written and spoken as well as the target of general audiences and children. Moreover, performance under the GlobalModel, the LocalModel, and the TransferModel does not always show the same best corpus. Corpus characteristics do not have a stable nor definitive influence on the performance of predicting the basic level. Nevertheless, the basic level can still be learned from the corpus characteristics.

With different sizes, intuitive thinking is that performance would be better with a larger corpus. However, it is not the case in concept frequencies from the corpora for predicting the basic level. There is always some unexpected drop in performance when the size goes larger. The good news is that a larger size of one frequency source could perform more reliable results with low variances. This finding keeps the same as Henry's (10). Performance goes up with a decreasing variance in the TransferModel and the GlobalModel is a good sign to predict the basic level in a large-scale hierarchy.

When comparing the corpus characteristics considering the size, we can discover some patterns and relationships from the tests in Section 6.1.2. In most cases, performance is improved with the corpus size increased using written corpora and general audience corpora in the LocalModel and the TransferModel. However, spoken corpora and corpora specific

to children do not have this characteristic. Moreover, children corpora even perform worse with a larger size in the GlobalModel and the TransferModel. Comparing under the same size, although it is hard to say which type of corpora performs better in the LocalModel, we can see written corpora tend to perform better than spoken ones and corpora to general audiences do better than ones specific to children in the LocalModel and the TransferModel. Especially, with a larger size, the performance of written corpora to general audiences stands out and is significantly better than that of the other corpus characteristics. That is why we adopt Google Books Ngram Corpus as the source of concept frequencies.

7.1.2 Google Books Ngram Corpus

As discussed, Google Books Ngram Corpus is a suitable source of concept frequency features considering the size because it is a written corpus whose target is general audiences. The time period when calculating frequencies in Google Ngram can stand for an accumulated amount of tokens varying years which contain the same effect as the size of a corpus. From the results in Section 6.2, it is evident that the performance of predicting the basic level is improved by the frequency features from Google Ngram. That is to say, we can effectively learn the basic level from a large general written corpus.

Analyzing the time period, neither Cohen's kappa nor balanced accuracy has a firm dependency on the time period. It implies that the frequency features from Google Ngram can be used without especially considering the corpus size. Additionally, the size of Google Ngram in one year is large enough and there is a factor that habits of using language change over time for humans. We find that the time period of the recent 50 years and 100 years resulted in the best performance among the three models. The aggregation methods of mean and maximum do not matter too much in most cases. Only the GlobalModel in the experiments, the maximum of frequencies performs better than the mean. Conducting feature selection and comparing the importance of each feature, we do not find which aggregation stands out. It remains further research to study the frequency aggregation in Google Ngram.

With the best combination of the features, frequency, as one of the synthetic features, can help to learn the basic level and improve the performance, as shown in Table 6.14. The frequency features from Google Ngram enhance Cohen's kappa approximately by 10% compared to the baseline.

7.2 Generative Semantic Features

SQ2: What new features concerning semantics can be generated to help improve the performance of predicting the basic level?

There are word embeddings and cues, which contain semantics, generated to be used as features to predict the basic level. From the results in Section 6.3, both of them after feature extractions are effective to improve the performance of predicting the basic level with the benchmark Random Forest classifier.

Word embeddings by Word2vec implicitly represent the semantics of concepts in their vectors. The method of extracting concept distances can make the semantics explicitly by the similarity between a concept and its direct hypernym(s). Using this method, we can obtain the distance between the basic level and its direct hypernym . We can then learn the basic level from these distances. Compared to the baseline in Table 6.16, word embeddings with the distance-based feature extraction can improve the performance. Frustratingly, there is only a little improvement with word embeddings. This distance can definitively represent some semantic properties of the basic level. However, the basic level concepts have few of these semantic properties in common and are less distinctive from non-basic level concepts'. The result for the LocalModel shows the least improvement which implies the semantic distance is less significant even within the same domain. It indicates that using the distance to represent semantics from word embeddings only has a limited contribution to predicting the basic level.

Cues generated by the fine-tuned BART are used to calculate the cue validity. It combines the concept's semantics and the definition of the basic level. By the same fine-tuned model, generation quality and criteria are the same for cues of different concepts. According to Section 4.4.2, the basic level concepts have the highest cue validities in a hierarchy. We can learn the basic level from the cue validity. Specifically, the range or the threshold of cue validities of the basic level concepts can be learned by the Random Forest classifier. From Table 6.17, the cues generated and the cue validity as the semantic features can effectively improve the performance of predicting the basic level. Performance improves the most in the LocalModel because cues of the basic level concepts in a specific domain tend to share the common features, hence, cue validities of them are similar within the same domain. Concepts across domains help to improve the generalization of the classifier. Therefore, improvement in the GlobalModel and the TransferModel is huge. Cohen's kappa in the TransferModel exceeds 0.7 for the first time which is even greater than some of the best performance with other methods including the baseline. Moreover, the growth rate

in the TransferModel suggests a good generalization ability of the cue generation and the cue validity which gives the confidence to detect the basic level in a large-scale hierarchy.

7.3 Large-Scale Detection

SQ3: How much would corpus characteristics and synthetic features improve basic level detection in a large-scale hierarchy?

Dataset of predicted basic level concepts under *entity.n.01* in WordNet is compared with the related research in Section 6.4. There are many concepts in common predicted as the basic level between our dataset and theirs. It indicates that most of the basic level concepts detected by the corpus characteristics and the synthetic features are agreed upon. Although it might be easy to compare our method to others with the same annotated gold standard of the basic level in a small dataset, there is no standard of the basic level for validation in WordNet. We are not able to confirm the concepts are correctly predicted to be the basic level. Despite that, the basic level effects would be mostly discovered with widespread agreement (10). Our basic level dataset is relatively small but covers most of the concepts in agreement with Hollink et al.'s and Henry's. It turns out that the method in this thesis has good sensitivity to detect the basic level.

Furthermore, the concepts that are agreed to be the basic level by multiple methods or datasets have more possibilities of being basic level concepts. Therefore, our dataset, Hollink et al.'s and Henry's data can be used together to decide whether a concept is the basic level, like three annotators. It is available to reuse for future research or development on KOS applications and the Semantic Web.

7.4 Limitations

When comparing the corpus characteristics, KBNC, a written corpus specific to children, has a relatively small size. It shows good performance but we do not have larger corpora for children in written discourse. Plus, the size of BNC is around a hundred times bigger than the size of KBNC. These make the comparison and analysis of the target audience not convincing sufficiently even performing the Wilcoxon rank-sum test with Bonferroni correction. The ideal frequency sources for the experiments would be corpora with different discourse types and target audiences in the same or similar total size. Using these corpora can avoid the dilemma in Section 6.1.2.4 that the lack of children-specific corpora in a larger size.

Acquisition of concept frequencies from Google Ngram Viewer is a time-consuming procedure for regular sleeping while posting requests. Although we designed an automatic crawler with the optimal sleeping policy, it still spends 10 days to obtain and aggregate the frequencies of 74,374 concepts in WordNet. There needs a more efficient approach to have frequency features from Google Ngram Viewer in terms of feature acquisition, selection, and aggregation. Otherwise, finding and validating a large written and general audience corpus would be an alternative approach to save processing time.

No gold standard for validating the predictions of the basic level in WordNet makes it hard to prove how much exactly corpus characteristics and the synthetic features improve basic level detection in a large-scale hierarchy. It has been found that the basic level effects are observed when the agreement of the basic level is universal (10). However, we only have two datasets of predicted basic level concepts. The agreement among the three predictions might be not significant to make sure to predict the basic level correctly because these three machine learning-based methods might have common defects. Therefore, it would be better to have a comparison with the predicted basic level dataset under *entity.n.01* in WordNet.

7.5 Future Work

The methods and findings provided in this thesis can help to develop an automatic basic level detection system for KOS applications and lay the foundation for further research on predicting the basic level. The future work can be imagined in aspects of machine learning for a better classifier and a group of applications potentially using the basic level.

With regard to machine learning, the binary classification task of predicting whether a concept is the basic level or not could upgrade to a multi-class task. Some practical classification criterion can be formalized, such as a concept being the basic level, a superordinate concept, or a subordinate concept. The prediction model could further be a regression model which can give a score to evaluate how much probability one concept is the basic level. This will be a quantitative method to tell how confident humans would regard a concept as the basic level. The methods for predicting the basic level only perform relatively well on training sets and validation sets but not robustly on a large scale with many domains in the real world. The generalization of models remains to be improved. Otherwise, the basic level theory and its benefits are difficult to be implemented in a production environment.

In terms of applications, basic level detection has a good prospect as a system of predicting the basic level from knowledge graphs and is used to set up better cognitive applications. Introduced the basic level theory and concepts into the field of computer vision, traditional image repositories, like ImageNet, could enhance their content-based image processing functions. As for Semantic Web, basic level concepts could help ontology to organize linked data and to better perform word sense disambiguation with knowledge graphs. They can provide a better user-machine interface and a cognitive interaction based on machine-readable techniques.

8

Conclusion

The Thesis studies corpus characteristics and synthetic features in predicting the basic level. The corpus characteristics include the discourse type, target audience, and size of a corpus. Reflecting on the first sub-research question, they do affect the performance of predicting the basic level. Considering the size of the corpus, written corpora to general audiences tend to perform the best, especially in a larger size. In addition, the frequency features from Google Books Ngram Corpus further improve the performance. Answering the second sub-research question, the distance-based features from word embeddings and cues generated by the fine-tuned BART, which belong to the semantic features, can improve the performance of predicting the basic level. Moreover, the cue validity of a concept calculated from the generated cues enhances the performance to a large extent.

We also propose a method to learn the basic level from synthetic features and to detect basic level concepts in WordNet. The synthetic features consist of structural, frequency, and semantic features. Compared to similar research and its predictions, our method shows good sensitivity to detect the basic level in a large-scale hierarchy. However, to answer the third sub-research question, it requires a gold standard of the basic level in WordNet, which as far as we notice does not exist yet. Instead, we compare our results with two datasets of predicted basic level concepts and can regard the concepts agreed upon in two datasets as the basic level. The basic level concepts predicted by our method are made available for future study.

With the three sub-research questions, we can answer the main research question. We can learn the basic level from natural language corpora with various characteristics and synthetic features. Moreover, the semantic features as one of the synthetic features contribute much to improving the performance of basic level detection.

References

[1] Yiwen Chen and Simone Teufel. **Synthetic Textual Features for the Large-Scale Detection of Basic-level Categories in English and Mandarin**. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8294–8305, 2021. viii, 2, 4, 5, 11, 12, 24, 45

[2] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. **Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension**. *arXiv preprint arXiv:1910.13461*, 2019. viii, 9, 10

[3] George A Miller. **WordNet: a lexical database for English**. *Communications of the ACM*, **38**(11):39–41, 1995. viii, 11, 13, 18

[4] Erin M Buchanan, Kathrene D Valentine, and Nicholas P Maxwell. **English semantic feature production norms: An extended database of 4436 concepts**. *Behavior Research Methods*, **51**(4):1849–1863, 2019. viii, 17, 23, 26, 28, 29

[5] Eleanor Rosch, Carolyn B Mervis, Wayne D Gray, David M Johnson, and Penny Boyes-Braem. **Basic objects in natural categories**. *Cognitive psychology*, **8**(3):382–439, 1976. 1, 2, 3, 4, 20, 21, 22, 24, 28

[6] Abebe Rorissa and Hemalata Iyer. **Theories of cognition and image categorization: What category labels reveal about basic level theory**. *Journal of the American Society for Information Science and Technology*, **59**(9):1383–1392, 2008. 1

[7] Wen-hao Chen, Yi Cai, Ho-fung Leung, and Qing Li. **Context-aware basic level concepts detection in folksonomies**. In *International Conference on Web-Age Information Management*, pages 632–643. Springer, 2010. 1, 12

[8] Steve Legrand. **Word sense disambiguation with basic-level categories**. *Advances in Natural Language Processing. Ed. Alexander Gelbukh, Research in Computing Science*, **18**:71–82, 2006. 1

[9] Panqu Wang and Garrison W Cottrell. **Basic level categorization facilitates visual object recognition**. *arXiv preprint arXiv:1511.04103*, 2015. 1

[10] Niamh Henry. *Learning the basic level from text: Studying different corpus characteristics in predicting the basic level.* Master's thesis, 2021. 2, 11, 19, 20, 21, 23, 36, 37, 48, 51, 52

[11] Roger Brown. **How shall a thing be called?** *Psychological review*, **65**(1):14, 1958. 3

[12] Lala Hajibayova. **Basic-level categories: A review**. *Journal of Information Science*, **39**(5):676–687, 2013. 3

[13] George Lakoff. *Women, fire, and dangerous things: What categories reveal about the mind.* University of Chicago press, 2008. 3

[14] Rebecca Green. **Vocabulary alignment via basic level concepts**. *Final Report*, 2003. 3, 47

[15] David J. Finton. *Cognitive-Economy Assumptions for Learning*, pages 626–628. Springer US, Boston, MA, 2012. 4

[16] Lee Roy Beach. **Cue probabilism and inference behavior.** *Psychological Monographs: General and Applied*, **78**(5-6):1, 1964. 6

[17] Lee Roy Beach. **Recognition, assimilation, and identification of objects.** *Psychological Monographs: General and Applied*, **78**(5-6):21, 1964. 6

[18] Stephen K Reed. **Pattern recognition and categorization**. *Cognitive psychology*, **3**(3):382–407, 1972. 6

[19] Leo Breiman. **Random forests**. *Machine learning*, **45**(1):5–32, 2001. 7

[20] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. **SMOTE: synthetic minority over-sampling technique**. *Journal of artificial intelligence research*, **16**:321–357, 2002. 7, 8

[21] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. **Distributed representations of words and phrases and their compositionality**. *Advances in neural information processing systems*, **26**, 2013. 8

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. **Efficient estimation of word representations in vector space**. *arXiv preprint arXiv:1301.3781*, 2013. 8

[23] Robyn Speer, Joshua Chin, and Catherine Havasi. **ConceptNet 5.5: An Open Multilingual Graph of General Knowledge**. pages 4444–4451, 2017. 8

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. **Attention is all you need**. *Advances in neural information processing systems*, **30**, 2017. 9

[25] Chad Mills, Francis Bond, and Gina-Anne Levow. **Automatic identification of basic-level categories**. In *Proceedings of the 9th Global Wordnet Conference*, pages 298–305, 2018. 10

[26] Laura Hollink, Aysenur Bilgin, and Jacco van Ossenbruggen. **Predicting the basic level in a hierarchy of concepts**. In *Research Conference on Metadata and Semantics Research*, pages 22–34. Springer, 2020. 10, 14, 19, 20, 23, 36, 47

[27] Marco Baroni and Alessandro Lenci. **Distributional memory: A general framework for corpus-based semantics**. *Computational Linguistics*, **36**(4):673–721, 2010. 11

[28] Paul Jaccard. **The distribution of the flora in the alpine zone. 1**. *New phytologist*, **11**(2):37–50, 1912. 12

[29] Mark Gluck. **Information, uncertainty and the utility of categories**. In *Proc. of the Seventh Annual Conf. on Cognitive Science Society, 1985*, 1985. 12

[30] Saul Albert, LE De Ruiter, and JP De Ruiter. **CABNC: The Jeffersonian transcription of the spoken British national corpus**, 2015. 15

[31] Brian MacWhinney. *The CHILDES project: Tools for analyzing talk, Volume II: The database*. Psychology Press, 2014. 15

[32] BNC Consortium et al. **British national corpus, XML edition**. *Oxford Text Archive. http://hdl. handle. net/20.500*, **12024**:2554, 2007. 15

[33] Brian MacWhinney. **Tools for analyzing talk part 1: The chat transcription format**. *Carnegie.[Google Scholar]*, 2017. 15

[34] Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K Gray, Google Books Team, Joseph P Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, et al. **Quantitative analysis of culture using millions of digitized books**. *science*, **331**(6014):176–182, 2011. 15

[35] Yuri Lin, Jean-Baptiste Michel, Erez Aiden Lieberman, Jon Orwant, Will Brockman, and Slav Petrov. **Syntactic annotations for the google books ngram corpus**. In *Proceedings of the ACL 2012 system demonstrations*, pages 169–174, 2012. 16

[36] Martin F Porter. **Snowball: A language for stemming algorithms**, 2001. 17

[37] Gregory L Murphy and Edward E Smith. **Basic-level superiority in picture categorization**. *Journal of verbal learning and verbal behavior*, **21**(1):1–20, 1982. 20

[38] Allan M Collins and Elizabeth F Loftus. **A spreading-activation theory of semantic processing.** *Psychological review*, **82**(6):407, 1975. 23

[39] Matt Post. **A Call for Clarity in Reporting BLEU Scores**. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. 27

[40] Jacob Cohen. **A coefficient of agreement for nominal scales**. *Educational and psychological measurement*, **20**(1):37–46, 1960. 30

[41] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M Buhmann. **The balanced accuracy and its posterior distribution**. In *2010 20th international conference on pattern recognition*, pages 3121–3124. IEEE, 2010. 30

[42] Henry B Mann and Donald R Whitney. **On a test of whether one of two random variables is stochastically larger than the other**. *The annals of mathematical statistics*, pages 50–60, 1947. 32

[43] Carlo Bonferroni. **Teoria statistica delle classi e calcolo delle probabilita**. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze*, **8**:3–62, 1936. 37

Appendix

A Results in EX_F_1: Structural Features

Train On	Test On	Cohen's Kappa	Balanced Accuracy
All 5 domains	All 5 domains	0.677	0.839

Table 8.1: Results of EX_F_1-GlobalModel

Train On	Test On	Cohen's Kappa	Balanced Accuracy
edible fruit	*edible fruit*	0.865	0.928
hand tool	*hand tool*	0.815	0.918
musical instrument	*musical instrument*	0.571	0.779
furniture	*furniture*	0.427	0.725
garment	*garment*	0.525	0.771
Average		0.640	0.825

Table 8.2: Results of EX_F_1-LocalModel

Train On	Test On	Cohen's Kappa	Balanced Accuracy
edible fruit, musical instrument, furniture, garment	*hand tool*	0.715	0.881
hand tool, musical instrument, furniture, garment	*edible fruit*	0.809	0.917
hand tool, edible fruit, furniture, garment	*musical instrument*	0.532	0.760
hand tool, edible fruit, musical instrument, garment	*furniture*	0.304	0.746
hand tool, edible fruit, musical instrument, furniture	*garment*	0.244	0.708
Average		0.521	0.802

Table 8.3: Results of EX_F_1-TransferModel

B Results in EX_W_x: Wilcoxon Rank-Sum Test

B.1 EX_W_0

B.1.1 GlobalModel

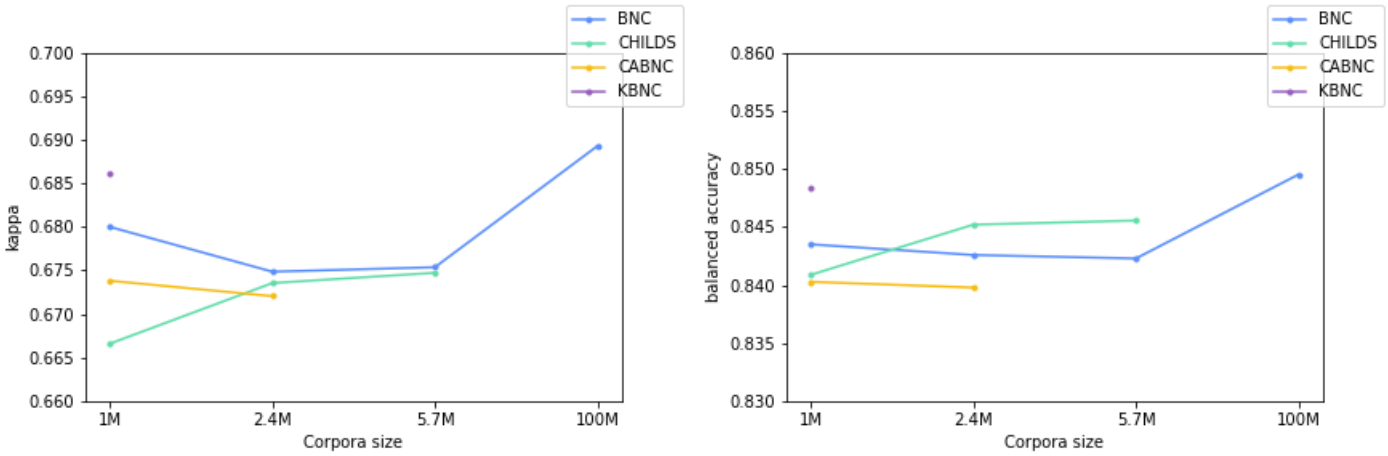

Figure 8.1: Averaged Metrics of EX_W_0-GlobalModel

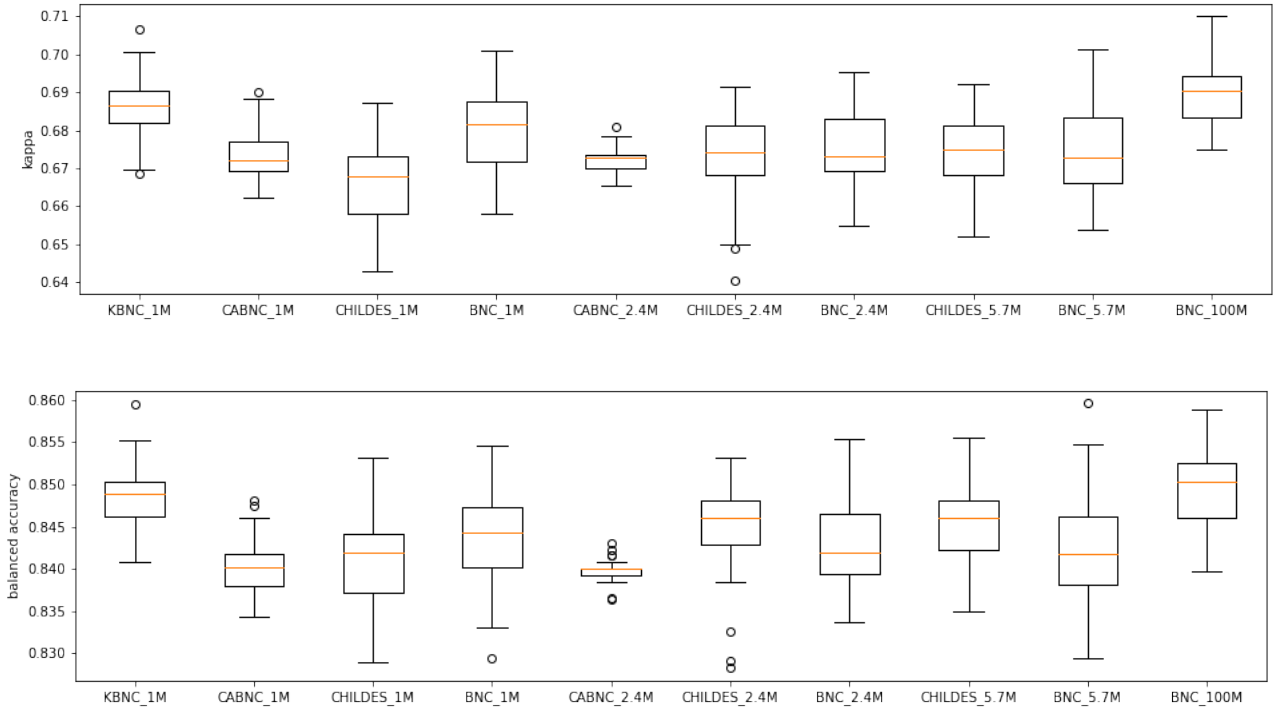

Figure 8.2: Overall Performance of EX_W_0-GlobalModel

Corpus	1M	2.4M	5.7M	100M
BNC	0.843	0.843	0.842	0.850
CHILDES	0.841	0.845	0.846	
CABNC	0.840	0.840		
KBNC	0.848			

Table 8.4: Averaged Balanced Accuracy of EX_W_0-GlobalModel

B.1.2 LocalModel

Corpus	1M	2.4M	5.7M	100M
BNC	0.833	0.836	0.837	0.842
CHILDES	0.830	0.832	0.833	
CABNC	0.829	0.833		
KBNC	0.827			

Table 8.5: Averaged Balanced Accuracy of EX_W_0-LocalModel

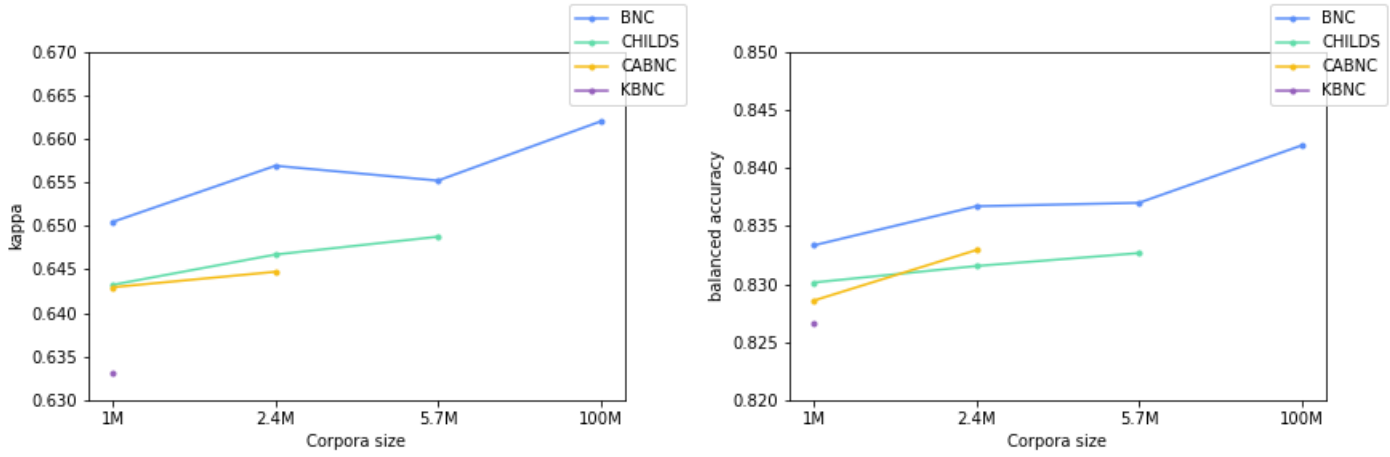

Figure 8.3: Averaged Metrics of EX_W_0-LocalModel

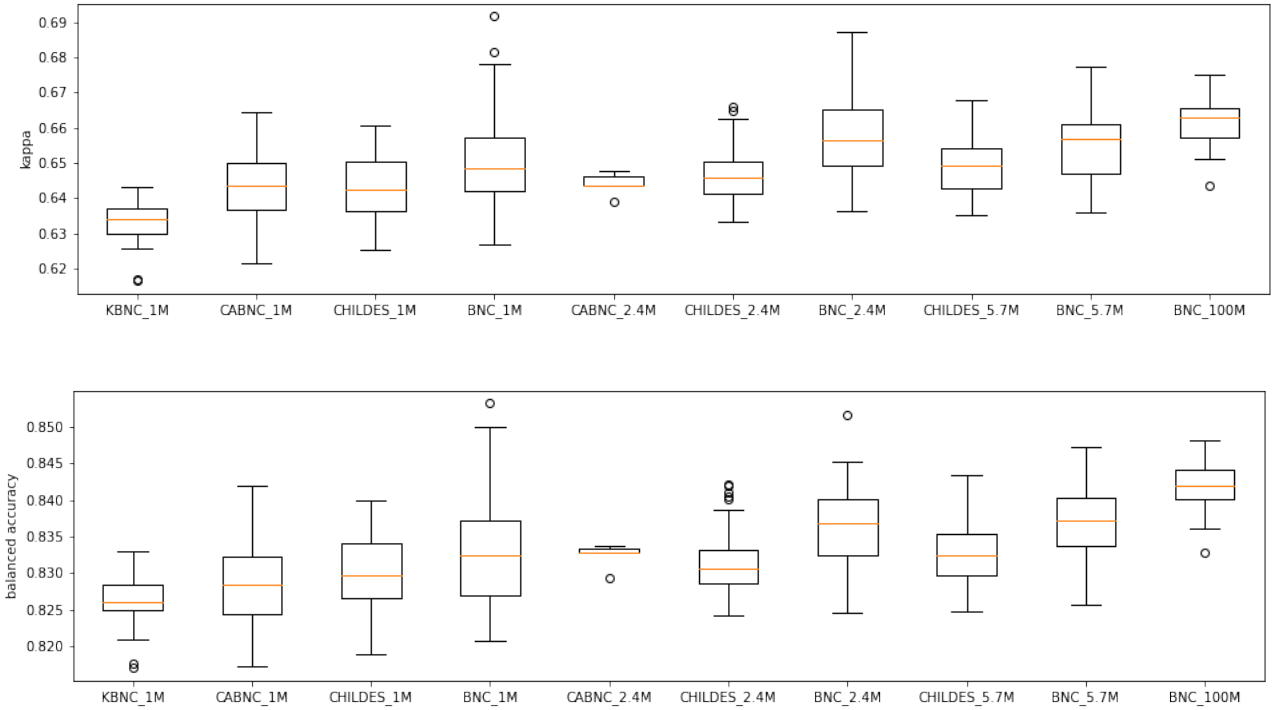

Figure 8.4: Overall Performance of EX_W_0-LocalModel

B.1.3 TransferModel

Corpus	1M	2.4M	5.7M	100M
BNC	0.811	0.815	0.816	0.814
CHILDES	0.809	0.808	0.806	
CABNC	0.796	0.799		
KBNC	0.810			

Table 8.6: Averaged Balanced Accuracy of EX_W_0-TransferModel

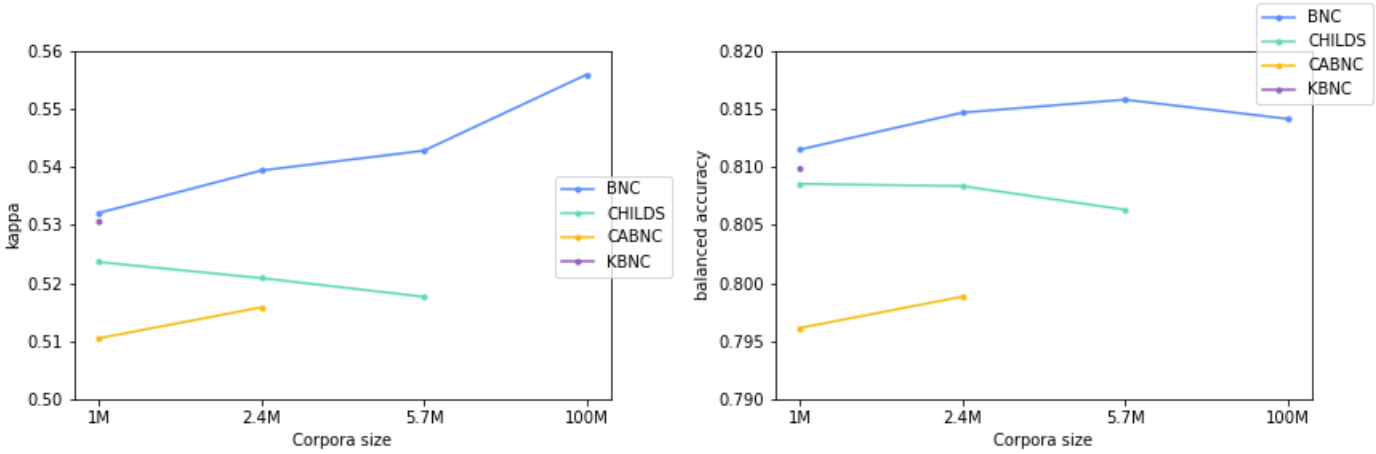

Figure 8.5: Averaged Metrics of EX_W_0-TransferModel

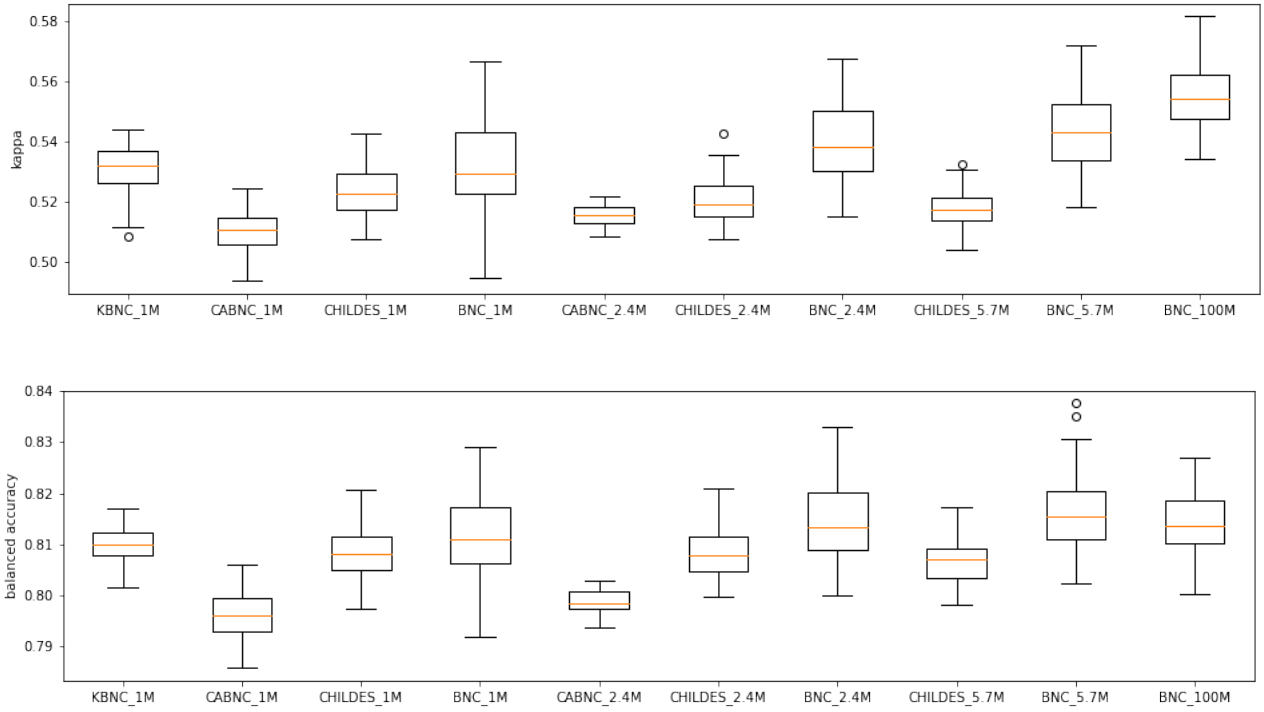

Figure 8.6: Overall Performance of EX_W_0-TransferModel

B.2 EX_W_1

Corpus Size	CABNC	CHILDES	BNC
1M-2.4M	0.885	0.000	0.817
1M-5.7M		0.000	0.859
1M-100M			0.000
2.4M-5.7M		0.534	0.609
2.4M-100M			0.000
5.7M-100M			0.000

Table 8.7: P-values of EX_W_1-GlobalModel on Balanced Accuracy

Corpus Size	CABNC	CHILDES	BNC
1M-2.4M	0.000	0.114	0.002
1M-5.7M		0.010	0.001
1M-100M			0.000
2.4M-5.7M		0.066	0.337
2.4M-100M			0.000
5.7M-100M			0.000

Table 8.8: P-values of EX_W_1-LocalModel on Balanced Accuracy

Corpus Size	CABNC	CHILDES	BNC
1M-2.4M	0.000	0.609	0.033
1M-5.7M		0.974	0.006
1M-100M			0.041
2.4M-5.7M		0.957	0.237
2.4M-100M			0.533
5.7M-100M			0.831

Table 8.9: P-values of EX_W_1-TransferModel on Balanced Accuracy

B.3 EX_W_2 EX_W_3

Corpus Size	Written	Spoken	General	Children
1M-2.4M	1.000	0.007	0.919	0.223
1M-5.7M	1.000	0.000	0.347	0.226
1M-100M	0.000		0.000	
2.4M-5.7M	0.609	0.000	0.109	0.534
2.4M-100M	0.000		0.000	
5.7M-100M	0.000		0.000	

Table 8.10: P-values of EX_W_2 and EX_W_3-GlobalModel on Balanced Accuracy

Corpus Size	Written	Spoken	General	Children
1M-2.4M	0.000	0.000	0.000	0.000
1M-5.7M	0.000	0.000	0.000	0.000
1M-100M	0.000		0.000	
2.4M-5.7M	0.337	0.004	0.002	0.066
2.4M-100M	0.000		0.000	
5.7M-100M	0.000		0.000	

Table 8.11: P-values of EX_W_2 and EX_W_3-LocalModel on Balanced Accuracy

Corpus Size	Written	Spoken	General	Children
1M-2.4M	0.000	0.104	0.012	0.000
1M-5.7M	0.000	0.000	0.000	0.000
1M-100M	0.001		0.000	
2.4M-5.7M	0.237	0.001	0.000	0.957
2.4M-100M	0.533		0.000	
5.7M-100M	0.831		0.831	

Table 8.12: P-values of EX_W_2 and EX_W_3-TransferModel on Balanced Accuracy

B.4 EX_W_4

Corpus Characteristics	1M	2.4M	5.7M
Written - Spoken	0.000	0.345	0.998
General - Children	1.000	1.000	0.998

Table 8.13: P-values of EX_W_4-GlobalModel on Balanced Accuracy

Corpus Characteristics	1M	2.4M	5.7M
Written - Spoken	0.465	0.000	0.000
General - Children	0.004	0.000	0.000

Table 8.14: P-values of EX_W_4-LocalModel on Balanced Accuracy

Corpus Characteristics	1M	2.4M	5.7M
Written - Spoken	0.000	0.000	0.000
General - Children	1.000	0.983	0.000

Table 8.15: P-values of EX_W_4-TransferModel on Balanced Accuracy

C Results of Frequency Feature From Google Ngram in EX_F_2

C.1 GlobalModel

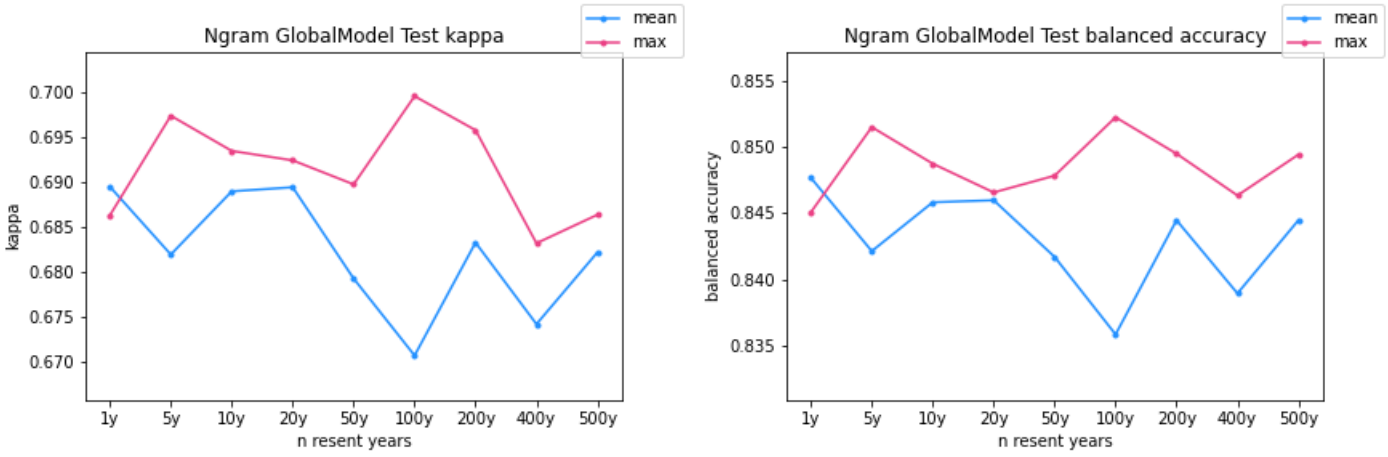

Figure 8.7: GlobalModel Performance of Frequency Features from Google Ngram

The best performing frequency feature setting from Google Ngram in GlobalModel is *50 year maximum*, *100 year maximum*, *400 year mean*, and *500 year maximum*. The importance in the Random Forest classifier is as follows:

Feature	Importance
normalized depth	0.355
normalized gloss length	0.142
shortest lemma length	0.106
normalized # part-whole relations	0.076
500 year maximum	0.075
100 year maximum	0.054
50 year maximum	0.044
# total hyponyms	0.044
maximal # polysemies	0.040
400 year mean	0.033
# lemmas	0.028
# direct hypernyms	0.002

Table 8.16: Importance of the Features in GlobalModel

C.2 LocalModel

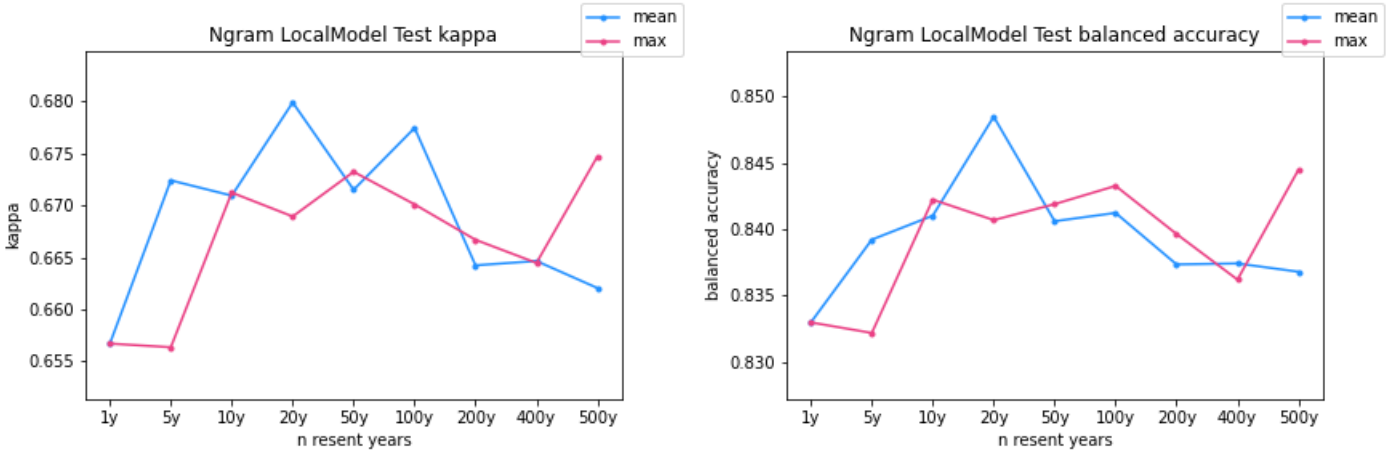

Figure 8.8: LocalModel Performance of Frequency Features from Google Ngram

The best performing frequency feature setting from Google Ngram in LocalModel is *100 year maximum* and *400 year mean*. The importance in the Random Forest classifier is as follows:

Feature	Importance
normalized depth	0.264
shortest lemma length	0.173
normalized gloss length	0.166
# total hyponyms	0.165
100 year maximum	0.089
maximal # polysemies	0.052
400 year maximum	0.047
# lemmas	0.030
normalized # part-whole relations	0.011
# direct hypernyms	0.003

Table 8.17: Importance of the Features in LocalModel

C.3 TransferModel

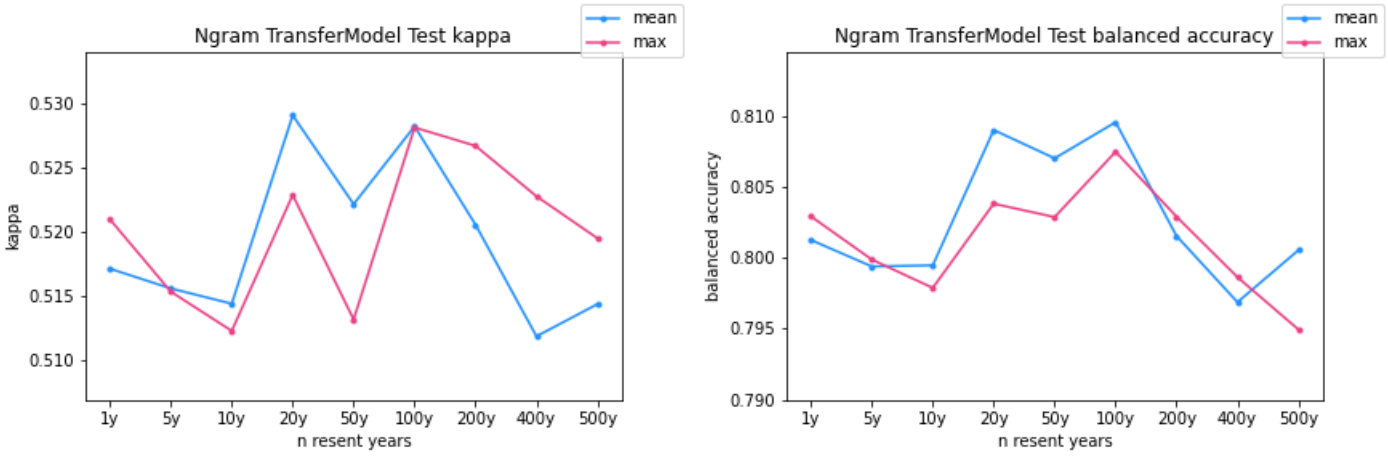

Figure 8.9: TransferModel Performance of Frequency Features from Google Ngram

The best performing frequency feature setting from Google Ngram in TransferModel is *5 year mean*, *50 year mean*, *100 year mean*, *200 year maximum*, and *500 year maximum*. The importance in the Random Forest classifier is as follows:

Feature	Importance
normalized depth	0.349
normalized gloss length	0.143
shortest lemma length	0.115
500 year maximum	0.078
normalized # part-whole relations	0.067
200 year maximum	0.046
maximal # polysemies	0.043
100 year mean	0.036
50 year mean	0.035
# total hyponyms	0.031
5 year mean	0.030
# lemmas	0.025
# direct hypernyms	0.002

Table 8.18: Importance of the Features in TransferModel