Ukrainian Catholic University

Bachelor Thesis

Data-Driven Approach to Automated Hypernym Hierarchy Construction for the Ukrainian WordNet

---

Author:
Nataliia Romanyshyn

Supervisors:
MSc. Dmytro Chaplynskyi and
MSc. Mariana Romanyshyn

A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science

in the

Department of Computer Sciences and Information Technologies
Faculty of Applied Sciences

Lviv 2023

Declaration of Authorship

I, Nataliia Romanyshyn, declare that this thesis titled, “Data-Driven Approach to Automated Hypernym Hierarchy Construction for the Ukrainian WordNet” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"*A language without a WordNet is at a severe disadvantage.*"

Maciej Piasecki, Stanisław Szpakowicz, Bartosz Broda

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Bachelor of Science

Data-Driven Approach to Automated Hypernym Hierarchy Construction for the Ukrainian WordNet

by Nataliia Romanyshyn

Abstract

WordNet is a valuable resource in the field of linguistics and natural language processing, providing a structured and comprehensive list of lexico-semantic relations among words in a language. Automatic approaches for constructing and expanding WordNets are gaining popularity due to the high cost associated with manual taxonomy creation. Unfortunately, the existing work on constructing the Ukrainian WordNet has been limited in scale and availability to the public, and it primarily focused on manual creation. This thesis aims to create a basis for the Ukrainian WordNet automatically, focusing on hypo-hypernym relations, which reflect the hierarchical structure of WordNet. The presented approach leverages the linking between Princeton WordNet (PWN) and Wikidata and multilingual resources from Wikipedia, which allowed to map 17% of PWN to Ukrainian Wiki. Three strategies for generating candidate words to fill the gaps in the constructed WordNet basis are proposed, including machine translation, the Hypernym Discovery model, and Hypernym Instruction-Following LLaMA. The latter model achieves high-performance results on the selected MOC metric (41.61%). Finally, an annotation tool is developed to enable lexicographers to review and edit the candidates generated by our methods to improve the coherence of the Ukrainian WordNet. Overall, this work is an important step towards bridging the WordNet gap in the Ukrainian language. With the proposed approach that combines automated techniques with expert human input, we provide a reliable basis for creating Ukrainian WordNet resource.

Acknowledgements

I am deeply grateful to my supervisors, Dmytro Chaplynskyi and Mariana Romanyshyn, for their invaluable guidance and insightful feedback that have been instrumental in shaping this thesis. I thank them for suggesting such a fascinating and scalable topic and for all the Friday calls that were filled not only with my thesis questions but also with a lot of other exciting topics.

I also want to express my gratitude to my family and friends, who have always been there for me, providing me with the love and emotional support that helped me to stay focused and motivated. A special thanks to my boyfriend, who was my first reader and always provided valuable feedback.

I am grateful to the Armed Forces of Ukraine, who gave me the opportunity to pursue my studies and write this bachelor's thesis during the ongoing Russo-Ukrainian war. Their sacrifice and dedication to our country have been an inspiration to me.

Finally, I would like to thank the entire team of the Faculty of Applied Sciences for their hard work in ensuring that education in Ukraine is of a high standard. I would like to say a special thank you to our teacher of Programming Fundamentals, Andriy Romaniuk, who sparked my interest in Computer Science and Natural Language Processing while I was still a philology student, and now he is protecting all of us and our country on the front line.

Contents

Declaration of Authorship i

Abstract iii

Acknowledgements iv

1 Introduction 1
- 1.1 Motivation 1
- 1.2 Focus of the Work 1
- 1.3 Goals of the Bachelor Thesis 2
- 1.4 Structure of the Thesis 2

2 Background Information 3
- 2.1 What is a WordNet? 3
- 2.2 WordNet Structure 3
- 2.3 Lexico-Semantic Relations 3

3 Related Work 6
- 3.1 Overview of WordNets 6
  - 3.1.1 WordNets in Other Languages 6
  - 3.1.2 Global WordNet Association 8
  - 3.1.3 Ukrainian WordNet: Status and Challenges 8
- 3.2 From Building Taxonomies to Hypernym Discovery 9
- 3.3 SemEval-2018 Task 9: Hypernym Discovery 10
- 3.4 Hypernymy with LLMs 10

4 Proposed Approach 11
- 4.1 PWN and Wikidata 12
- 4.2 Gap Ranking 14
- 4.3 Methods for Generating Candidates to Fill the Gaps 14
  - 4.3.1 Machine Translations 15
  - 4.3.2 Hypernym Discovery Model 16
  - 4.3.3 LLaMA 17
- 4.4 Evaluation Metrics 17
- 4.5 Annotation tool 18

5 Experiments 20
- 5.1 WordNet Basis 20
- 5.2 Ukrainian adaptation to SemEval-2018 Task 9 21
  - 5.2.1 Data Collection 21
    - The Corpus 21
    - Vocabulary 22
    - Input Terms and Gold Hypernyms 22
    - Statistics 23

5.2.2 Experimental Setup 23
5.2.3 Comparison of Embeddings 24
Baseline 24
Fasttext Embeddings 24
5.2.4 Results 24
5.3 Instruction-Following LLaMA 25
5.3.1 Instructions Generation 25
Lean 25
Full 26
Multiple 26
5.3.2 Results 26
5.4 Error Analysis 27

6 Conclusions **30**
6.1 Contribution 30
6.2 Limitations 30
6.3 Future Work 31

Bibliography **32**

List of Figures

2.1 Example entry in Princeton WordNet 3.1. The WordNet's definition of itself including its hypernym, hyponyms, and instance hyponym. 4
2.2 An example of the hypernym, hyponyms, and co-hyponyms hierarchy. Image taken from Commons (2017). 5

3.1 A comparison of the highest level noun hierarchy between Princeton WordNet 3.0 and the Ukrainian WordNet-like dictionary. Adapted from Kulchytsky, Romaniuk, and Khariv (2010). 9

4.1 Proposed approach for building a Ukrainian WordNet through linking Princeton WordNet with Wikidata and Wikipedia, using a Gap Ranking algorithm to identify best gaps and generate candidate terms through automatic translation, Hypernym Discovery, and Hypernymy Instruction-Following LLaMA. Professional linguists will further manually annotate the automatically generated candidates. 11
4.2 Pipeline for building the basis of the Ukrainian WordNet utilizing the linking between Princeton WordNet, Wikidata, and the Ukrainian Wikipedia. 12
4.3 The tree construction for the synset 11482925-n in PWN using the pipeline from Figure 4.2. The green indicates that the corresponding page in the Ukrainian Wikipedia was found, and its title was extracted. Red indicates that the Ukrainian title could not be retrieved, so a lemma from the PWN synset was temporarily preserved in place of the gap. 13
4.4 Example of applying the algorithm for finding the best nodes for filling. The tree fragment for synset 00871261-n is shown, with the best path marked with a green arrow. Filling the node *"arithmetic operation"* is more effective than the other nodes, as it produces five new hypernym-hyponym pairs compared to only one pair for other nodes. 15
4.5 Main page of the gap annotation system displaying a list of gaps and candidate translations. 19
4.6 Annotation interface for the gap annotation system, showing gap background information and candidate translations. 19

5.1 Ukrainian Hypernym Discovery dataset statistics, categorized by its type and in total. Training and test sets are split equally; trial refers to the development set. 23
5.2 Metrics comparison of the two top-performing models based on MOC score for all entity types. The models are HD_Fasttext_bin and LoRA_Hypernymy_Full. 28

List of Tables

4.1 Comparison examples of gap translations obtained using machine translation methods. All terms are nouns. The gap is identified as the most optimal for filling using the algorithm described in Section 4.2. 16

5.1 General statistics related to the development of the Ukrainian WordNet basis, including the total number of synsets in the PWN3.1, the number and percentage of synsets linked to Wikidata and the Ukrainian Wikipedia. 20
5.2 Ukrainian Hypernymy Pairs dataset statistics. This table presents the number of word pairs obtained for each type of relationship, including hypernym-hyponym, co-hyponyms, hypernym-instance, and co-instances. 21
5.3 Statistical information overview of UberText2.0, including its subcorpora, time span, number of sources, texts, and tokens. Source: Chaplynskyi (2023) 22
5.4 Our system's performance on the test set for the adapted Ukrainian version of the SemEval 2018 Task 9. HD_Baseline refers to the Hypernym Discovery model with word2vec embeddings, HD_Fasttext_bin to the one using binary fasttext, and HD_Fasttext_vec utilized plain-text fasttext vectors. The best score for each model is marked in **bold**. 25
5.5 The LoRA fine-tuning results with hypernymy instructions using different setups. The "LoRA_Hypernymy_Lean" setup only uses the most basic hypernymy instructions, while "LoRA_Hypernymy_Full" includes 19 instruction patterns for a single input query. In the "Multiple" setup, three relation types (hypernym, hyponym, and co-hyponym) were used in addition to diverse patterns. 27
5.6 Examples of predictions made by Hypernym Discovery model HD_Fasttext_bin, showing input hyponym (query), its type, overlap coefficient (OC), and top-6 predicted hypernyms for high and low score. The OC of 100 indicates that all ground truth hypernyms were predicted, and 0 means that the model proposed no relevant options. Bolded examples are correct hypernyms. 28
5.7 Examples of predictions made by the LoRA_Hypernymy_Full model with their corresponding query types, OC scores, and predicted hypernyms. The examples are divided into two sections: high OC examples, where the model performed well, and low OC examples, where the model's predictions were less accurate. 29

List of Abbreviations

CILI	**C**ollaborative **I**nter**L**ingual **I**ndex
DFS	**D**epth-**F**irst **S**earch
GWA	**G**lobal **W**ordNet **A**ssociation
IR	**I**nformation **R**etrieval
KUI	**K**nowledge **U**nifying **I**nitiator
LLM	**L**arge **L**anguage **M**odels
LoRA	**Lo**w-**R**ank **A**daptation
MAP	**M**ean **A**verage **P**recision
MOC	**M**ean **O**verlap **C**oefficient
MRR	**M**ean **R**eciprocal **R**ank
NLP	**N**atural **L**anguage **P**rocessing
NLTK	**N**atural **L**anguage **T**ool**K**it
OMW	**O**pen **M**ultilingual **W**ordnet
OOV	**O**ut-**o**f-**V**ocabulary
P@k	**P**recision at **k**
POS	**P**art **o**f **S**peech
PWN	**P**rinceton **W**ord**N**et
R-P	**R**-**P**recision
RDF	**R**esource **D**escription **F**ramework

Dedicated to my family and the Armed Forces of Ukraine

Chapter 1

Introduction

1.1 Motivation

WordNet is a valuable resource, providing a structured and comprehensive list of the lexical and semantic relationships among words in a language. It is a highly versatile tool used by linguistics, psychology, and natural language processing (NLP) professionals. Its applications include word sense disambiguation, machine translation, information retrieval, automatic text classification, and summarization (Morato et al., 2004).

Unlike a traditional thesaurus, WordNet organizes words based on their semantic and lexical relations to other concepts. Therefore, it is a valuable tool for disambiguating word senses in NLP. Measuring word similarity is another common application of WordNet, and such a measure can be used in spell checking or question answering.

Over the past years, researchers focusing on languages other than English have attempted to address the lack of digital lexical databases like the Princeton WordNet (Miller, 1994). Given the high cost associated with manual taxonomy creation, there has been an increasing interest in automatic approaches for constructing and expanding WordNets.

Several studies have shown that this method can be used to create and extend WordNets for different languages, such as Chinese (Wang and Bond, 2013), Arabic (Elkateb et al., 2006), and Urdu (Adeeba and Hussain, 2011). In these studies, multilingual resources were used to identify and extract synsets and semantic relations, which were then used to automatically construct WordNets with varying degrees of manual validation and refinement.

Nevertheless, there is a lack of such a lexical database for Ukrainian. The existing work on constructing the Ukrainian WordNet has primarily focused on manual creation. However, these efforts have been limited in scale and availability to the public. To address these limitations, this thesis builds upon previous work by introducing a technique that leverages multilingual resources extracted from Wikidata[1] and Wikipedia[2] to create a basis for the Ukrainian WordNet.

1.2 Focus of the Work

Creating a WordNet with all its semantic relations, including synonymy, antonymy, hypo-hypernymy, and meronymy, requires extensive resources and expertise. Therefore, for the purpose of this thesis, we have decided to focus specifically on hypo-hypernym relations. It is a fundamental type of semantic relation that reflects the hierarchical structure of WordNet. By focusing on hypo-hypernymy, we aim to create a solid basis for the Ukrainian WordNet that can be expanded to include other semantic relations in the future.

---

[1] https://www.wikidata.org/wiki/Wikidata:Main_Page

[2] https://www.wikipedia.org

1.3 Goals of the Bachelor Thesis

1. Introduce a novel technique for creating a basis of Ukrainian WordNet by leveraging multilingual resources extracted from Wikidata and Wikipedia.

2. Develop an algorithm that maps Ukrainian Wikipedia titles to synsets in the Princeton WordNet and finds hyponyms for each synset to construct a hierarchical tree diagram.

3. Develop a method of prioritizing the gap nodes in the Ukrainian WordNet that would create the most links when filled in.

4. Propose strategies for automated generation of candidate words to fill the gaps, namely translating English lemmas into Ukrainian using machine translation and building two models that generate candidates given the gap hyponym — Hypernym Discovery and Hypernymy Instruction-Following LLaMA.

5. Develop an annotation tool that enables lexicographers to review and edit the candidates generated by the proposed approach.

1.4 Structure of the Thesis

The remaining work is divided into the following chapters.

In Chapter 2, we review the main concepts and terms used in this paper: what is a WordNet, its structure, and lexico-semantic relations.

Chapter 3 presents an overview of various WordNets. First, we discuss the WordNets of other languages, both monolingual and multilingual. Then we define the main challenges and status of building a Ukrainian WordNet. Lastly, a description of the Hypernym Discovery task and discussions of key approaches to solving it will be presented.

In Chapter 4, we describe in detail the pipeline of our approach: compiling the basis for the Ukrainian WordNet utilizing existing resources and methods for filling the gaps.

Chapter 5 presents statistics of the datasets obtained using the methodology described in the previous section and introduces the main experiments performed for the Hypernym Discovery task and instruction-tuned LLaMA. Furthermore, we discuss and analyze the obtained results.

Finally, Chapter 6 summarizes the work done, discusses our approach's limitations, and suggests future work directions.

Chapter 2

Background Information

2.1 What is a WordNet?

A WordNet is a lexical database that serves as a valuable language resource for various natural language processing applications. It contains a vast collection of terms that are grouped into synsets, i.e., sets of synonyms. Each synset expresses a distinct concept. The synsets are interlinked through lexico-semantic relations.

Unlike a traditional dictionary, WordNet organizes words and concepts based on their meanings rather than in alphabetical order. It uses a semantic network structure to represent the relationships among words, making it a more complex resource for NLP tasks. While WordNet shares similarities with a thesaurus in its approach to categorizing words according to their meanings, it differs from a thesaurus in several significant aspects. Firstly, WordNet links not only different forms of a word but also specific word senses. This means that words that are found close to each other in WordNet are semantically disambiguated. Secondly, WordNet labels the semantic relations among words, which is not present in a traditional thesaurus.

The first and most well-known WordNet is the Princeton WordNet, which began as a psychological experiment to explain how lexical meaning is stored in mind and how children acquire lexical meaning. Nevertheless, it has become increasingly popular among NLP scholars dealing with the meaning of words and their relations and among ontology experts.

An ontology is a collection of facts about a particular domain of knowledge or reality. Basile (2015) argues that WordNet can be seen as a lightweight ontology about words, senses, and a series of relations among them. However, additional work is necessary to transform it into a formal ontology specified in some logic formalism (Gangemi et al., 2003).

2.2 WordNet Structure

The basic building block of a WordNet is *synset* or synonym set, which refers to words that hold similar meanings and can be used interchangeably in many contexts (Vossen, 2002). Synsets are connected to each other through conceptual-semantic and lexical relations (see Section 2.3). Each synset also includes a brief definition or gloss, as well as one or more example sentences that illustrate the usage of the words in the synset. If a word has multiple meanings, each distinct meaning is represented in a separate synset, making each form-meaning pair in WordNet unique. WordNet consists of four sub-nets, one for each part of speech (POS) — nouns, verbs, adjectives, and adverbs. The general structure of PWN entry is illustrated in Figure 2.1.

2.3 Lexico-Semantic Relations

The following are the core semantic relations that define synsets in WordNet:

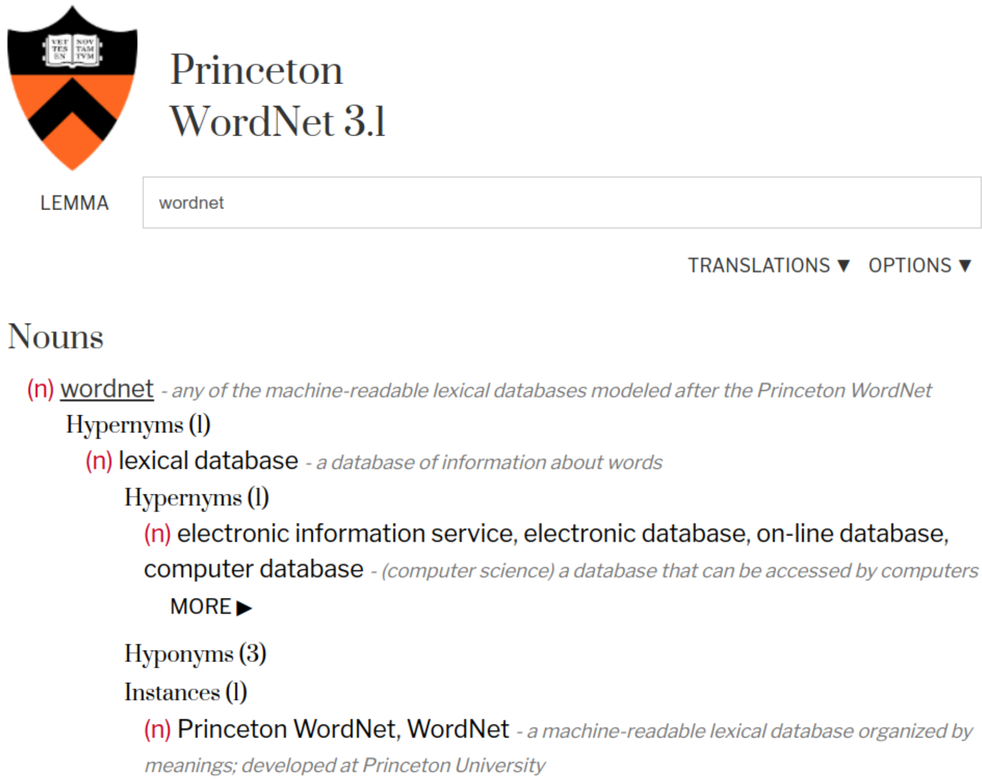

Figure 2.1: Example entry in Princeton WordNet 3.1. The WordNet's definition of itself including its hypernym, hyponyms, and instance hyponym.

1. The **super-subordinate**, or **hyponymy and hypernymy**, or **IS-A relation**, is the central relation for nouns that shapes the hierarchical structure of the WordNet. It links general synsets to more specific ones. For example, *rose* is a hyponym of *flower*, which is a hypernym of *rose*. Words that are hyponyms of the same hypernym are called co-hyponyms. Figure 2.2 presents an example of a two-level hierarchy. The hypernymy relation is:

   (a) Asymmetric: substitute X and Y in the sentence X *is a kind of* Y and determine if it makes sense. For example, *A missile is a kind of weapon* makes sense, but *A weapon is a kind of missile* — not.

   (b) Transitive: if X is a hyponym of Y, and Y is a hyponym of D, then X is a hyponym of D. For example, *begonia* is a hyponym of *flower*, and *flower* is a hyponym of a *plant*; therefore, *begonia* is a hyponym of a *plant*.

   WordNet hypernymy includes:

   (a) Types — common nouns: *armchair* is a type (hyponym) of *chair*;

   (b) Instances — specific persons, countries, and geographic entities: *Valerii Zaluzhnyi* is an instance (instance hyponym) of a *commander-in-chief*. Instances are always leaf nodes in their hierarchies.

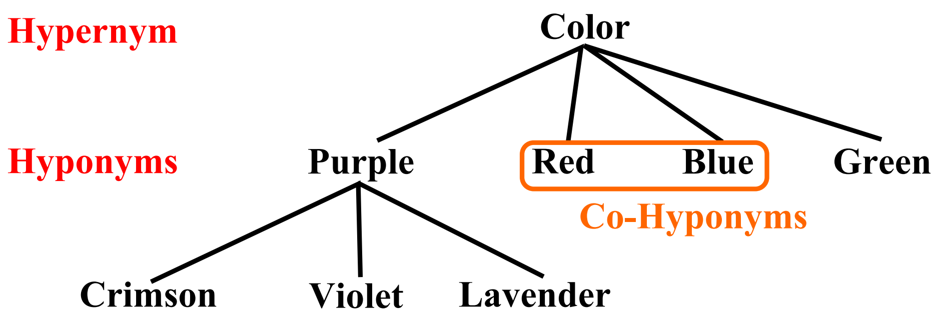

Figure 2.2: An example of the hypernym, hyponyms, and co-hyponyms hierarchy. Image taken from Commons (2017).

2. **Meronymy** is a semantic relation that describes the **whole-part** or **HAS-A** relation between a meronym (part) and a holonym (whole). For example, a *finger* is a holonym (part) of a *hand*, which is its meronym (whole). The meronymy relation is not transitive; parts are inherited only from their superordinates, not backward.

3. **Antonymy and synonymy** show semantic contract and similar relations, respectively. While antonymy defines the oppositeness between words — *day* and *night*, synonymy refers to words similar in meaning — *car* and *automobile*.

Furthermore, there are **entailment**, which connects verbs based on their logical relationships, and **attribute** relations — which connect nouns to adjectives that describe their attributes.

Domain/Usage, **Cause**, **Verb Group**, and **Similar To** are additional relations that are not as commonly used as the primary ones but still provide valuable semantic information.

In this work, we focus solely on the hyponym and hypernym relations; other types of semantic relations are out of the scope of this thesis and are left for further research.

Chapter 3

Related Work

3.1 Overview of WordNets

The Princeton WordNet of the English language is the first and by far the best developed WordNet (Miller, 1994). It has, in fact, become the standard and is commonly used as a reference for other WordNets and WordNet-related work. The PWN[1] includes more than 117,000 synonym sets and provides a rich set of relations. The success of PWN has inspired the creation of WordNets for other languages, which have helped to advance natural language processing research in various fields, such as natural language generation (Jing, 1998), metaphor detection (Mao, Lin, and Guerin, 2018) or textual entailment (Lan and Jiang, 2018).

To build a WordNet for other languages from scratch, two main approaches are commonly used in the literature: merge and expand (Vossen, 1998).

The merge approach involves creating a semantic network specific to the language being studied and then linking its synsets with those of the Princeton WordNet in the final stage of the project. The expand approach involves mapping or translating local words directly to the synsets of an existing WordNet.

The expand approach is more efficient and requires less linguistic knowledge, but it may result in less accurate representations of the language's semantic and lexical structure. However, many WordNet developers select this approach due to the universal structure of lexical semantics that exists across languages (Youn et al., 2016).

3.1.1 WordNets in Other Languages

Some notable examples of multilingual WordNets include:

- the EuroWordNet project[2], which created multilingual WordNets for several European languages: Dutch, Italian, Spanish, German, French, Czech, and Estonian (Vossen, 1997);
- the MultiWordNet project[3] that includes Italian, Spanish, Portuguese, Hebrew, Romanian, and Latin languages (Pianta, Bentivogli, and Girardi, 2002);
- the BalkaNet project[4], which created WordNets for several Balkan languages: Bulgarian, Greek, Romanian, Serbian, Turkish and extended the Czech WordNet previously developed in the EuroWordNet project (Tufis, Cristeau, and Stamou, 2004).

Among the various monolingual WordNets available, plWordNet (Piasecki, Szpakowicz, and Broda, 2009) holds a distinguished position. It was developed using a corpus-based approach, where a large text corpus served as the primary data source for all phases of

[1] https://wordnet.princeton.edu
[2] https://archive.illc.uva.nl//EuroWordNet
[3] https://multiwordnet.fbk.eu/english/home.php
[4] http://www.dblab.upatras.gr/balkanet

development. The process involved the systematic extraction of lemmas from the corpus for inclusion in the plWordNet and the automated acquisition of lexico-semantic relations for presentation to the editors. In addition to the corpus, dictionaries and encyclopedias were also used to supplement the language competence of the trained linguist editors who had the final say in all linguistic matters related to the plWordNet (Maziarz et al., 2016).

Developed since 2005, it has emerged as a comprehensive lexical database of the Polish language and currently boasts the title of being the largest lexico-semantic database in the world, with plWordNet 4.2[5] containing approximately 229,000 synsets. In addition to serving as an extensive resource for Polish language researchers, plWordNet can also function as a reliable Polish-English or English-Polish dictionary due to its connection with PWN (Rudnicka, Witkowski, and Piasecki, 2021).

Thoongsup et al. (2009) describes the process of constructing a Thai WordNet using a semi-automatic method based on the Princeton WordNet. The method involves aligning PWN synsets with an existing bilingual dictionary using English equivalents and their part-of-speech tags and then manually translating them as needed. A web-based collaborative workbench called Knowledge Unifying Initiator (KUI) is also developed to facilitate the revision of synset assignments and to provide a framework to create Asian WordNet via the linkage through PWN synsets.

Batsuren et al. (2019) used a combination of expert, monosemy, and hypernym-based translations to expand the Princeton WordNet and create a Mongolian version with over 23,000 synsets, 40,000 senses, and 26,000 words. The focus of expert translations is on finding the most appropriate words for the concept in terms of linguistic context use rather than on word-for-word translation between synsets. In monosemy translation, the algorithm checks for a one-to-one mapping between the lexical resource and the bilingual dictionary for the input word. If a match occurs, the corresponding synset and sense are assigned and added to the answer set. In hypernym-based translation, the algorithm iterates through all possible pairs of synset and sense and checks for the same part of speech. It then checks if the collocate noun of the sense is a hypernym of the synset in the lexical resource. If so, the pair is added to the answer set. According to a manual assessment of the resource, its quality was evaluated at 96.4%.

Siegel and Bond (2021) presented an open-sourced German WordNet OdeNet, which has been integrated into the natural language processing library NLTK (Bird, Klein, and Loper, 2009), thereby making it available to other researchers for utilizing in their own NLP projects. The first version of OdeNet was completely automatically created by combining existing resources, namely, the OpenThesaurus German synonym lexicon[6], the Open Multilingual WordNet (Bond and Paik, 2012; Bond and Foster, 2013), and the Princeton WordNet of English. Relations were added to OdeNet by utilizing the regularity of the hyponymy relationship to the head of German compounds. The resulting WordNet comprises approximately 120,000 lexical entries in about 36,000 synsets. About 20,000 of these synsets are linked to synsets in the English PWN and then to the multilingual CILI (Bond et al., 2016) numbers.

In general, the availability of WordNets in different languages has greatly facilitated natural language processing research and applications across various linguistic and cultural contexts. These resources have also provided a valuable foundation for further research in lexical semantics (Wei et al., 2015), word sense disambiguation (Siemiński, 2011), and other related areas.

[5] `http://plwordnet.pwr.edu.pl/wordnet`
[6] `https://www.openthesaurus.de`

3.1.2 Global WordNet Association

To maintain WordNet resources all over the world, in 2000 was founded a non-profit organization, the Global WordNet Association (GWA)[7]. The primary goal of the GWA is to enhance the dissemination of knowledge, facilitate communication among scholars who use WordNets, and synchronize the construction of new WordNets. Specifically, the GWA strives to establish and advocate for techniques, norms, and shared structures for creating new WordNets that can connect and exchange data. Currently, the GWA has over 70 registered WordNets[8], which have been constructed and are accessible or are in the process of development. Unfortunately, the Ukrainian language is not represented at this organization.

During the 3rd GWA Conference, the idea of creating a free worldwide WordNet grid was introduced. A consequence of this idea is the creation of the Open Multilingual WordNet (OMW)[9] by Bond and Paik (2012). The OMW project is a significant effort toward making WordNets accessible in multiple languages. This project's primary goal is to develop a shared format for WordNets and link them together to make them easily usable. OMW and its components are publically available and can be used, modified, and shared freely. Currently, OMW has two versions available:

1. OMW Version 1[10], which links hand-crafted and automatically created WordNets for over 150 languages via the PWN;

2. OMW Version 2[11] is an experimental version that employs the Collaborative Interlingual Index (CILI) to connect the WordNets (Bond et al., 2016).

Overall, the OMW project is a significant step towards making WordNets easily accessible in multiple languages, and it provides an important resource for researchers and developers.

3.1.3 Ukrainian WordNet: Status and Challenges

The first published works on the construction of the Ukrainian WordNet were carried out in the 2010s.

Kulchytsky, Romaniuk, and Khariv (2010) conducted a study that focused on analyzing the relationships between nouns in the Princeton WordNet, selecting core nouns for the Ukrainian language, and organizing them into a hierarchical structure (Figure 3.1). The resulting WordNet-like dictionary includes 194 synsets, of which 183 are interconnected by hypo-hypernymy, 14 by antonymy, and 150 by meronymy/homonymy. The research in question was conducted manually using frequency dictionaries. Unfortunately, the project was not continued, and the results were not made publicly available.

Anisimov et al. (2013) described the process of developing a lexical semantic database for the Ukrainian language called UkrWordNet. The article focuses on the research and development of automated techniques for replenishing and extending the UkrWordNet. The method developed for creating new nodes involved generating them from Ukrainian Wikipedia articles and binding them to the synsets of UkrWordNet. The paper also proposed a new measure of semantic similarity using latent semantic analysis (Deerwester et al., 1990) to improve the quality of the bindings. After manual post-processing, UkrWordNet contained over 82,000 synsets and approximately 145,000 nouns in the lexicon. Yet, the main drawback of this work is the lack of a resulting resource available to the public.

[7] `http://globalwordnet.org`
[8] `http://globalwordnet.org/resources/wordnets-in-the-world`
[9] `https://omwn.org/`
[10] `https://omwn.org/omw1.html`
[11] `https://omwn.org/omw2.html`

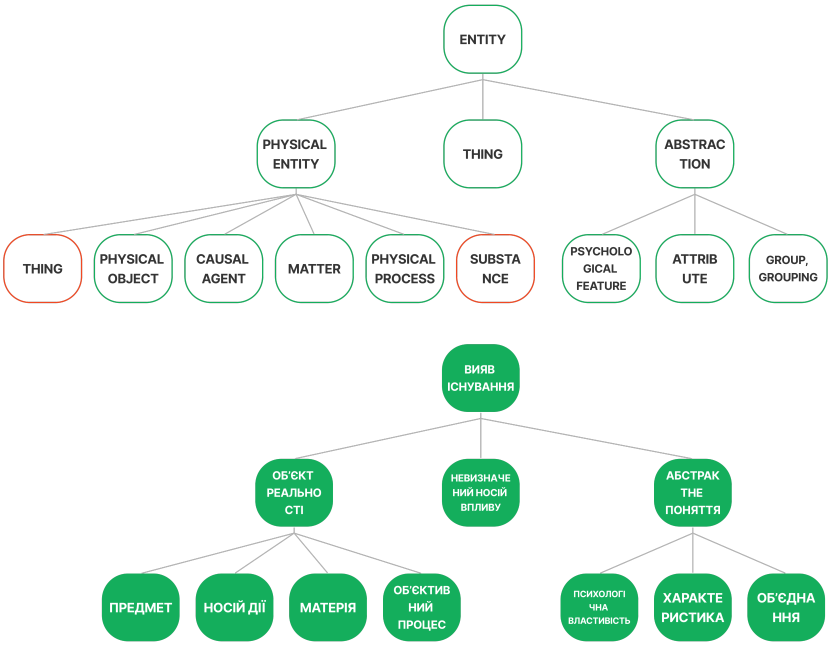

Figure 3.1: A comparison of the highest level noun hierarchy between Princeton WordNet 3.0 and the Ukrainian WordNet-like dictionary. Adapted from Kulchytsky, Romaniuk, and Khariv (2010).

Other developments in the field of the Ukrainian WordNet include materials[12] from theses of students of Lviv Polytechnic, but they are of a limited size.

Hence, the development of a WordNet for the Ukrainian language that has a representative number of relations and is open source remains an open field for research.

3.2 From Building Taxonomies to Hypernym Discovery

Camacho-Collados (2017) argue that the research focus has shifted from building taxonomies entirely or partly from scratch to hypernym relation extraction due to reduced complexity in collecting training data and a more straightforward evaluation process. Hypernymy detection (Weeds et al., 2014; Pannitto, Salicchi, and Lenci, 2017; Nguyen et al., 2017; Roller, Kiela, and Nickel, 2018) is easier to evaluate as it is measured with standard evaluation benchmarks. However, Levy et al. (2015), Santus et al. (2016), and Shwartz, Santus, and Schlechtweg (2017) indicate that supervised systems in hypernym detection task tend to display lexical memorization phenomena, which is attributed to the inherent modeling of the datasets (Camacho-Collados et al., 2018).

Therefore, the ultimate goal is to discover or find hypernyms for a given concept, which is the primary practical feature in downstream applications such as question answering (Prager, Radev, and Czuba, 2001). Unfortunately, the step from detecting hypernymy relations to discovering hypernyms is feasible but not trivial.

Camacho-Collados (2017) suggests that hypernym discovery (Espinosa-Anke et al., 2016) is a research field in itself and proposes constructing better benchmarks and developing methods for this task and points out that traditional information retrieval (IR) measures (Bian et al.,

[12] https://github.com/lang-uk/wordnet/tree/main/resources

2008) such as Mean Reciprocal Rank (MRR), Mean Average Precision (MAP), R-Precision (R-P), or Precision at k (P@k) can be used to evaluate these systems.

3.3 SemEval-2018 Task 9: Hypernym Discovery

To advance research in the field of hypernym discovery, *SemEval-2018 Task 9*[13] was organized (Camacho-Collados et al., 2018). It aimed to reformulate the hypernymy detection task by asking participants to discover suitable hypernyms from a target corpus given an input term. The authors provided a reliable framework for evaluating proposed models with the abovementioned IR metrics. Two types of data were given to participants for training their models: a vast unlabeled text corpus and a limited training dataset containing a query and its corresponding hypernyms. The competition had five subtasks that fell into two groups:

1. General-purpose Hypernym Discovery: English, Spanish, Italian;
2. English domain-specific Hypernym Discovery: Medicine, Music.

The best-performing systems were **CRIM** (Bernier-Colborne and Barrière, 2018) for English, Medicine, and Music, **300-sparsans_r1** (Berend, Makrai, and Földiák, 2018) for Italian, and **NLP_HZ** (Qiu et al., 2018) for Spanish.

We decided to adapt the *SemEval-2018 Task 9: Hypernym Discovery* setting for Ukrainian and reproduce Bernier-Colborne and Barrière (2018) approach, as their system exhibited superior performance over other systems and baselines. The model involves utilizing both supervised projection learning and unsupervised pattern-based hypernym discovery. The unsupervised part extends the basic pattern-based approach (Hearst, 1992) by identifying co-hyponyms and discovering additional hypernyms based on the assumption that most multi-word English expressions are compositional. The supervised part applies a decision function to all candidate hypernyms and selects the most likely candidates for a given query using projection learning. The word embeddings for all queries and candidates were learned using pre-tokenized corpora and the word2vec model (Mikolov et al., 2013). Finally, the hybrid approach takes the top 100 candidates according to each system, normalizes their scores, and sums them to rerank the candidates according to a new score.

We utilized only the supervised approach for our experiments, as the pattern-based one is not sharable across languages.

3.4 Hypernymy with LLMs

Pretrained large language models (LLMs) have demonstrated impressive results on various NLP tasks, motivating us to explore their potential for Hypernym Discovery. Prior work by Hanna and Mareček (2021) used a prompting methodology to investigate BERT's (Devlin et al., 2019) knowledge of hypernymy, revealing that while BERT has some understanding of hypernymy and outperforms other unsupervised models, its comprehension remains limited, particularly with uncommon hyponyms and hypernyms.

In this thesis, we aim to examine the potential of another state-of-the-art LLM, multilingual LLaMA (Touvron et al., 2023), which has demonstrated outstanding performance on different NLP benchmarks. The authors of LLaMA noted that fine-tuning this model on instructions leads to promising results and left it for future work. Therefore, instead of prompting, we decided to fine-tune LLaMA on hypernym instructions to investigate if it can perform hypernyms suggestions.

[13] https://competitions.codalab.org/competitions/17119

Chapter 4

Proposed Approach

The methodology described involves building a Ukrainian WordNet using the expand method with the Princeton WordNet as a pivot structure and linking to Wikidata and Ukrainian Wikipedia. The algorithm (Section 4.1) uses these three resources to construct a tree diagram by mapping Ukrainian Wikipedia titles to synsets in the PWN and finding hyponyms for each synset. The Gap Ranking algorithm (Section 4.2) is employed to identify the best gap nodes for filling. Then, we propose several strategies (Section 4.3) for generating candidate words to fill the gaps. The first uses English lemmas translated into Ukrainian with Google Translate, Bing, and DeepL. Secondly, we adapt the Hypernym Discovery task for Ukrainian and build a model which generates candidates given the gap hyponym. Finally, hypernym candidates are generated with the Instruction-Following LLaMA model. The candidates will be further reviewed and edited by lexicographers using an annotation tool (Section 4.5) developed with Payload CMS and MongoDB. Overall, the proposed approach combines automated techniques with expert human input to create a comprehensive and reliable resource for the Ukrainian language. Figure 4.1 summarizes the proposed methodology.

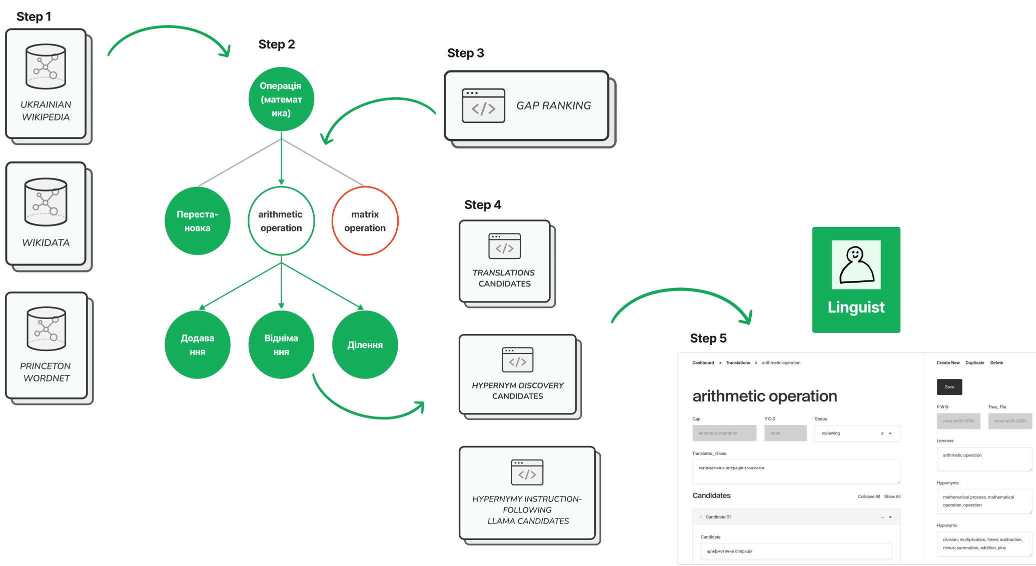

Figure 4.1: Proposed approach for building a Ukrainian WordNet through linking Princeton WordNet with Wikidata and Wikipedia, using a Gap Ranking algorithm to identify best gaps and generate candidate terms through automatic translation, Hypernym Discovery, and Hypernymy Instruction-Following LLaMA. Professional linguists will further manually annotate the automatically generated candidates.

4.1 PWN and Wikidata

Our approach benefits from the linking between PWN and Wikidata proposed by McCrae and Cillessen (2021). Wikidata is a free and open knowledge graph that provides structured and machine-readable data on various topics. The authors suggested the two-staged development of linking between PWN and Wikidata datasets. First, utilizing the hapax legomenon links; these are links for which there is only one sense for the lemma in PWN, and only one page in Wikidata has the lemma as the English title. The paper also explored how to extend the linking process utilizing NLP techniques. Human annotators validated the results of both stages.

Furthermore, we utilize one of the primary purposes of Wikidata — to serve as a central database for storing and managing structured data, which can be accessed and used by other Wikimedia[1] projects like Wikipedia. Wikidata's language-independent identifiers allow for the efficient management and linking of information across different language versions of Wikipedia articles, making it easier for users to access and navigate information in the preferred language (Vrandečić, 2013).

Hence, our algorithm uses three main resources: PWN, Wikidata, and Ukrainian Wikipedia. The general pipeline is summarized in Figure 4.2.

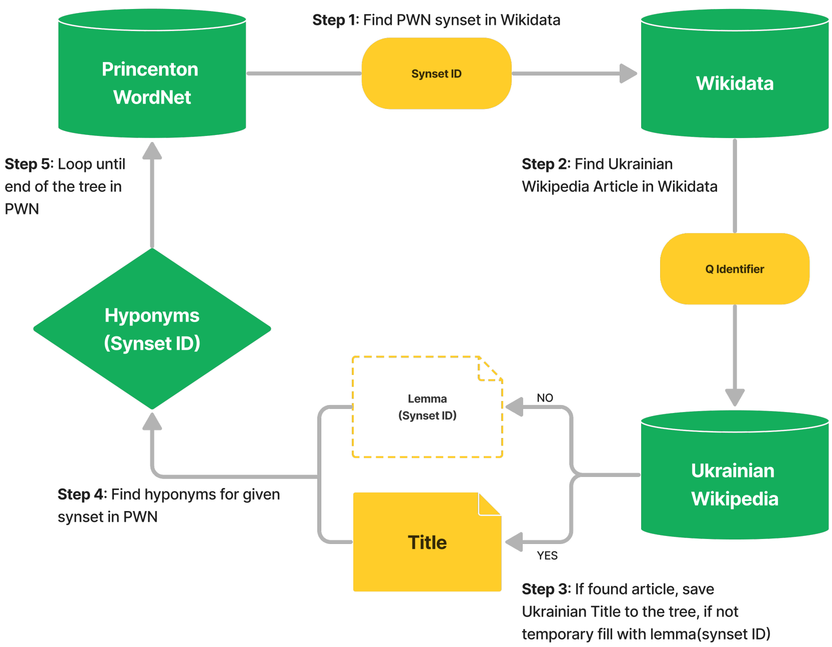

Figure 4.2: Pipeline for building the basis of the Ukrainian WordNet utilizing the linking between Princeton WordNet, Wikidata, and the Ukrainian Wikipedia.

The procedure is as follows:

1. Find the PWN synset that is linked with Wikidata using synset ID.
2. With Wikidata Q identifier[2] search Ukrainian Wikipedia article in Wikidata.
3. At this step, two options are possible:

[1] `https://www.wikimedia.org`
[2] `https://www.wikidata.org/wiki/Q43649390`

(a) If the search at the previous step was successful, we got the word to fill the node in our tree.

(b) If not, we temporarily store the lemma from PWN at this node to fill this lacuna later. We will discuss the gap-filling methods in Section 4.2 and Section 4.3.

4. Then, we find hyponyms for the given synset ID.

5. And, continue from step 1 until the end of the tree is reached for the given synset in PWN.

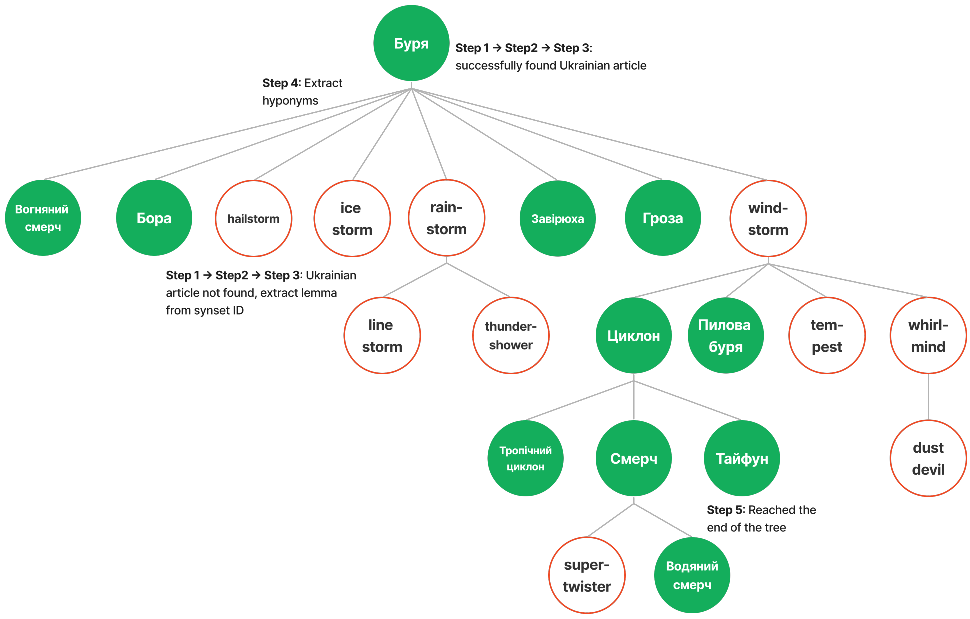

Figure 4.3: The tree construction for the synset 11482925-n in PWN using the pipeline from Figure 4.2. The green indicates that the corresponding page in the Ukrainian Wikipedia was found, and its title was extracted. Red indicates that the Ukrainian title could not be retrieved, so a lemma from the PWN synset was temporarily preserved in place of the gap.

Figure 4.3 presents the subtree construction for the synset 11482925-n in the Princeton WordNet using the proposed approach. The tree diagram illustrates the different steps of the pipeline, including the retrieval of pages from Ukrainian Wikipedia and the extraction of their titles. The green indicates that the corresponding page in the Ukrainian Wikipedia was found, and its title was successfully extracted. On the contrary, the red indicates that the Ukrainian title could not be retrieved, resulting in a gap in the tree. In this case, a lemma from the PWN synset was temporarily preserved in place of the missing title. Overall, the analysis of the illustrated subgraph from WordNet shows that 52% of the subgraph can be mapped to the Ukrainian Wikipedia through the proposed system.

The reasons why Ukrainian Wiki titles were not extracted include the following:

1. synset ID is not linked with Wikidata, which is the most common reason for missing pages;

2. the lack of a Ukrainian page on the wiki for the corresponding Wikidata Q identifier;

3. synset ID in Wikidata leads to an empty page;

Furthermore, we encountered several cases when the same Ukrainian Wikipedia title was obtained for both hypernym and its hyponym.

These factors highlight the challenges in integrating linguistic data from different sources into the new WordNet system, and addressing them would require careful and thorough analysis to ensure the accuracy and completeness of the WordNet.

Therefore, in the following sections, we will propose a method for finding the best nodes to fill in the subtree to make it more connected. Additionally, we will present several methods to automatically generate candidate words that can be used by the lexicographer in the process of gap annotation.

4.2 Gap Ranking

The Gap Ranking algorithm aims to identify the best gap nodes for filling, i.e., those with the most non-gap children in the given tree.

Let T be a tree with each node representing a synset in the Ukrainian WordNet, and the edge showing the hypernym-hyponym relation between synsets. The objective of the algorithm is to build a path P in T such that the number of valid pairs of nodes in P is maximized, where a *valid pair* is defined as a pair of nodes (u, v) such that u is a gap node and v is a non-gap child of u.

The algorithm uses a depth-first search (DFS) tree traversal approach to search for the optimal path P. Starting from the root node, it recursively traverses the tree, considering each node as a potential gap node. For each gap node, the algorithm computes the number of valid pairs of nodes in its subtree by considering its non-gap children. If the number of valid pairs exceeds the maximum number seen so far, the algorithm updates the maximum number of valid pairs and records the path leading to this node. Finally, the algorithm selects the path with the most valid pairs.

The metric the algorithm optimizes for is the number of valid pairs of nodes in the selected path P. This metric helps identify the gap nodes with the highest potential for enhancing the quality of the Ukrainian WordNet. With the Gap Ranking algorithm, we identified that by simply completing **793** gaps, we could obtain a total of **5403** new hyper-hyponym pairs.

Figure 4.4 demonstrates this method's application. It presents a tree fragment for synset 00871261-n. From this subgraph, it can be observed that filling the node *"arithmetic operation"* is more effective than, for instance, the *"construction"* node, as the former produces five new hypernym-hyponym pairs while the latter only produces one pair. The best path is marked with the green arrow.

4.3 Methods for Generating Candidates to Fill the Gaps

In this study, we employed two approaches to generate candidates to fill the gaps in our lexical resource.

The first approach involved using an English lemma of the synset and automatic translation methods to translate this word into Ukrainian. This method allowed us to quickly generate a list of potential candidates for the gap.

The second approach used the hyponym of the gap, which had a corresponding title in the Ukrainian Wikipedia article. We then utilized the Hypernym Discovery model and Instruction-Following LLaMA to generate hypernyms for this hyponym.

Using these two complementary methods, we generated a comprehensive list of candidates to fill the gaps in our WordNet. We believe this approach would allow annotators to quickly identify the most likely options for the missing nodes, reducing the time and effort required to annotate the resource manually.

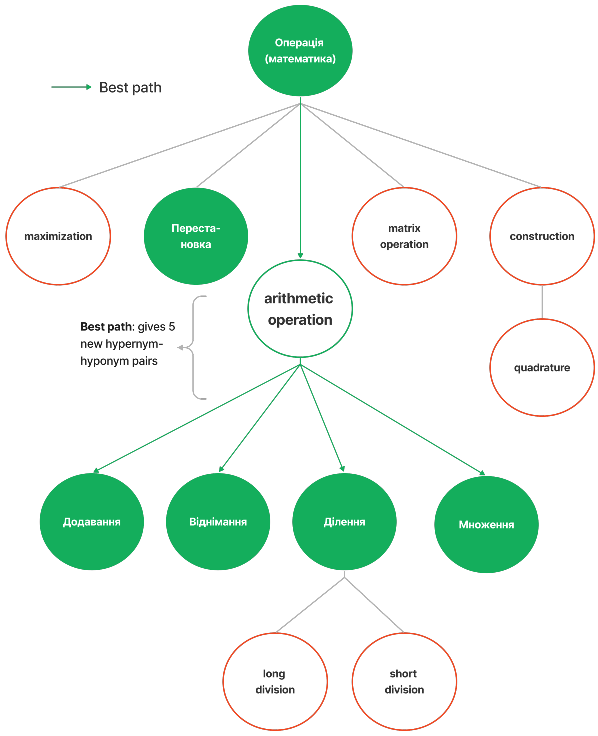

Figure 4.4: Example of applying the algorithm for finding the best nodes for filling. The tree fragment for synset 00871261-n is shown, with the best path marked with a green arrow. Filling the node *"arithmetic operation"* is more effective than the other nodes, as it produces five new hypernym-hyponym pairs compared to only one pair for other nodes.

4.3.1 Machine Translations

The automatic translations of the gaps included three main steps:

1. use the previously translated Princeton WordNet 3.1 into Ukrainian[3] with Google Translate (Johnson, 2012) and Bing (Liu, 2011). With the synset ID, we extracted the translation of the gap and added it to the list of candidates;

2. utilize the popular neural machine translation platform DeepL (Ronzon, 2018) to provide additional translations with two approaches:

   (a) DeepL Direct: involved feeding a single lemma into the DeepL translator through its Python API.

   (b) DeepL Contextualized: we created a sentence in the format of *"<Synset lemmas> is a <gloss>."* and fed it into the DeepL translator. The first lemma from the resulting translation was then extracted and recorded as a candidate for the gap in WordNet.

3. combine the output from the first two steps and save them as automatically translated candidates for the lacuna in WordNet.

[3] https://github.com/lang-uk/wordnet/tree/main/pwn_translated_basic

Overall, this approach allowed us to quickly generate a list of potential translations for the missing nodes. Table 4.1 shows a comparison of gap translations obtained with discussed methods.

Table 4.1: Comparison examples of gap translations obtained using machine translation methods. All terms are nouns. The gap is identified as the most optimal for filling using the algorithm described in Section 4.2.

Gap	DeepL Direct	DeepL Contextualized	Translated PWN3.1
crosspiece	перемичка	хрестовина	хрестовина
performance	продуктивність	вистава	вистава, спектакль
head cabbage	качанна капуста	качанна капуста	головна капуста
manual	інструкція	мануал	посібник
agency	агентство	агентство	офіс, орган

4.3.2 Hypernym Discovery Model

To perform Hypernym Discovery in the Ukrainian language, we adopted the setting provided for this task by Camacho-Collados et al. (2018). In Section 5.2.1, we provide a detailed overview of the data creation process. This section briefly presents the model used to run experiments for this task.

We utilized the supervised part of the model proposed by Bernier-Colborne and Barrière (2018), the SemEval-2018 Task 9 winners. Their approach uses pre-trained word embeddings and projection learning to discover the hypernyms of a given query (hyponym). The model consists of the following components:

1. Lookup Table: the model retrieves embeddings e_q and e_h for the given query q and candidate hypernym h from a lookup table. These embeddings are learned beforehand on a large unlabeled text corpus.

2. Projection Matrix (trainable): the embedding of the query is then multiplied by a 3-D tensor containing k square projection matrices ϕ_i for $i \in \{1, \ldots, k\}$, producing a matrix P containing the projections of e_q. The k is fixed to 24 in the reference paper. The weights of ϕ are initialized by adding random noise to an identity matrix.

3. Dot Product: the model checks how close each of the k projections of e_q are to e_h by taking the dot product: $s = P \cdot e_h$.

4. Affine Transformation (trainable): the column vector s is then fed to an affine transformation.

5. Sigmoid Activation: a sigmoid activation function is applied to the output of the affine transformation to obtain an estimate of the likelihood that q and h are related by hypernymy.

6. Rank Hypernyms: to discover the hypernyms of a given query, the model computes the likelihood $y = \sigma(W \cdot s + b)$ for all candidates and selects the top-ranked ones.

To train the model, the authors suggested employing negative sampling, which involves generating a fixed number of negative examples for each positive example of a query-hypernym pair in the training data by replacing the hypernym with a randomly drawn word from the vocabulary. The objective is to train the model to output a high likelihood (y) for positive examples and a low likelihood for negative examples. To accomplish this, the binary cross-entropy of the positive and negative training examples should be minimized. Specifically, for a given example, the binary cross-entropy is computed as follows:

$$H(q, h, t) = t \cdot \log(y) + (1 - t) \cdot \log(1 - y), \tag{4.1}$$

where q refers to a query, h is a candidate hypernym, t is the target, which takes a value of 1 for positive examples and 0 for negative ones, and y is the predicted likelihood. To optimize the model, the binary cross-entropy is summed for every example in the training set D (containing both the positive and negative examples), resulting in the cost function:

$$J = \sum_{(q,h,t) \in D} H(q, h, t) \tag{4.2}$$

4.3.3 LLaMA

LLaMA is a collection of transformer-based (Vaswani et al., 2017) language models ranging from 7B to 65B parameters. In their work, Touvron et al. (2023) emphasized that the model's performance should be improved by enlarging the amount of training data rather than the number of parameters. They argued that the main expense for LLMs comes from conducting inference on the trained model rather than the computational expense of the training phase.

The 65B parameter models were trained on a corpus of 1.4 trillion tokens, whereas the LLaMA 7B model was trained on 1 trillion tokens. The authors used publicly available data sources such as web pages scraped by CommonCrawl (67%), open source code repositories from GitHub, Wikipedia in 20 different languages, including Ukrainian, public domain books from Project Gutenberg, and questions with answers from Stack Exchange websites.

In this study, we utilized the pretrained LLaMA-7B that can be found on the Hugging Face model hub[4]. The model's hyperparameters were: 4096 dimensions, 32 heads and layers, a learning rate of $3.e-4$, 4M batch size, and 1T tokens. The fine-tuning details and strategies for instructions creation are discussed in Section 5.3.

4.4 Evaluation Metrics

Camacho-Collados et al. (2018) proposed evaluating the Hypernym Discovery systems as a soft ranking problem utilizing top N hypernyms produced by the model. The value of N was determined based on the maximum number of hypernyms found in the training and testing data for a single hyponym. Therefore, we set the value of N to 6 for our experiments. This ensures that the evaluation is done based on the realistic number of hypernyms that can be expected for each input term.

To evaluate the performance of the Hypernym Discovery models, the authors suggested utilizing several Information Retrieval metrics, including MAP, MRR, and P@k. In order to get a more thorough understanding of the model's ability to predict relevant hypernyms regardless of the order of predictions, we propose using the Mean Overlap Coefficient (MOC) as an additional evaluation criterion. The performance of the Hypernymy Instruction-Following LLaMA was measured with the same metrics. Let us briefly introduce each of them.

1. **Mean Reciprocal Rank** measures how well a system is able to rank the relevant hypernyms by rewarding the position of the first correct result in the ranked list of outcomes.

[4] https://huggingface.co/decapoda-research/llama-7b-hf

The formula is as follows:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}, \tag{4.3}$$

where Q represents the number of queries and $rank_i$ refers to the rank position of the first relevant outcome for the ith run.

2. **Mean Average Precision**:

$$MAP = \frac{1}{|Q|} \sum_{q \in Q} AP(q), \tag{4.4}$$

where $AP()$ is the average precision of each individual hypernym obtained from the search space. Essentially, MAP is the average of AP scores across all the queries in Q.

3. **P@k** measures the number of correctly retrieved hypernyms at different cut-off thresholds, specifically for k values of 1, 3, and 6 in our case.

4. **Mean Overlap Coefficient**:

$$MOC = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{|GT_i \cap P_i|}{|GT_i|} \tag{4.5}$$

where GT represents the set of ground truth hypernyms, and P represents the set of predicted ones for a given input term. The numerator calculates the number of hypernyms that are common between the ground truth and predicted sets, while the denominator ensures that the metric is normalized by the size of the ground truth set.

The formulas 4.4 and 4.3 were taken from Camacho-Collados et al. (2018).

We considered the MOC score the most helpful metric for our specific task — generating candidates for professional annotators, as it indicates the proportion of relevant values that were predicted regardless of their order.

Our work on Ukrainian WordNet is a continuous process, and we aim to obtain a larger dataset to retrain our models and improve their quality. We plan to iterate the process of identifying the best gaps to fill, getting candidate suggestions from the model, and manually checking the results until the model can generate relevant hypernyms in most cases. At that point, we will give more weight to the MRR metric, which takes the prediction rank into account and helps generate a relevant variant based on the order of the predicted element, reducing the need for manual work. However, due to time constraints, we limited our work to the candidate generation stage in this thesis and left the next steps for future research.

4.5 Annotation tool

To facilitate the manual annotation of gaps in the Ukrainian WordNet, we developed an annotation tool using the Payload CMS [5] and MongoDB [6]. The tool provides lexicographers with a user-friendly interface that allows them to review the candidates generated by the methods described above and choose the best option, add their own translations, or modify the proposed one. We chose to use the Payload CMS because of its flexibility and ease of use. The

[5] https://payloadcms.com/

[6] https://www.mongodb.com/

CMS is designed to manage content efficiently and provides a straightforward and intuitive interface for this purpose.

Figure 4.5 shows the first page of the system, which displays a list of gaps to be filled and the number of candidates generated for them. If you want to edit a specific one, there is also a field for searching for a gap by its lemma. The "Reviewing" and "Approved" buttons on the left sidebar will automatically filter the items by their review status.

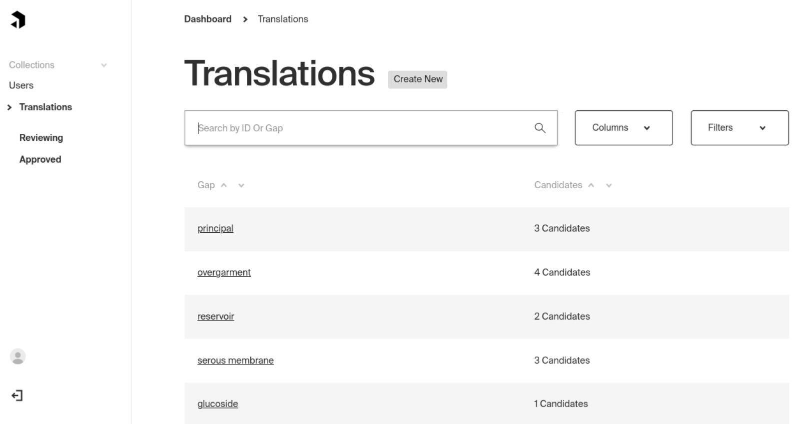

Figure 4.5: Main page of the gap annotation system displaying a list of gaps and candidate translations.

Clicking on the gap lemma leads to a page where the annotation is done. The interface is presented in Figure 4.6. It displays the gap's background information in the right sidebar, including part of speech, Princeton WordNet ID, gloss, lemmas, hypernyms, and hyponyms from PWN. Additionally, the tool includes a gloss from the automatically translated WordNet. The candidates section presents all generated translations; the annotator can review them, edit them as needed, or remove and add their options. The annotator can change the status of each gap from "reviewing" to "approved" when the section with candidates is processed. This feature allows the annotator to keep track of the progress of their work.

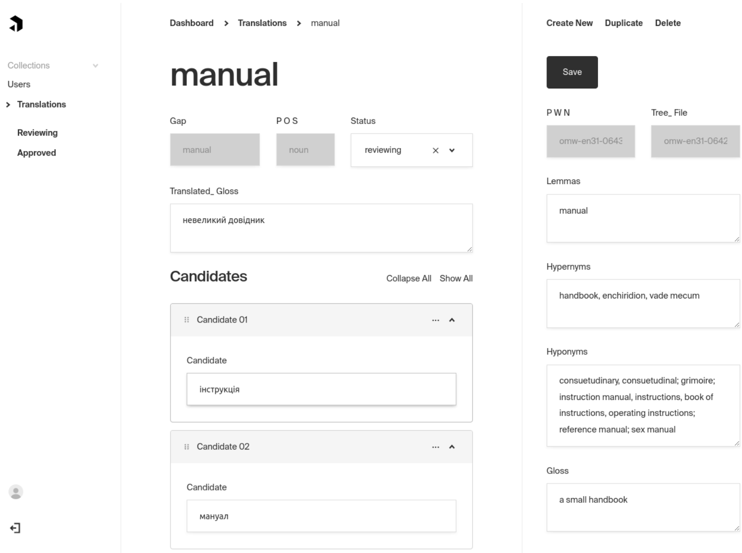

Figure 4.6: Annotation interface for the gap annotation system, showing gap background information and candidate translations.

Chapter 5

Experiments

5.1 WordNet Basis

To link the data from PWN, Wikidata, and Ukrainian Wikipedia, we implemented a Python (Van Rossum and Drake, 2009) scraper using the web-crawling framework Scrapy (Hoffman, Graña, and Olveyra, 2008), wtf_wikipedia library (Kelly, 2017) for Wikipedia parsing, wn package (Goodman and Bond, 2021), which provides an interface to WordNet data, and an RDF (Resource Description Framework) query language SPARQL (Prud'hommeaux and Seaborne, 2008).

We managed to link 17% of the Princeton WordNet, resulting in 21,015 synsets forming the foundation of the Ukrainian WordNet. Out of the 127,020 PWN3.1 synsets, we could link 23% to Wikidata; subsequently, 17% of those synsets were connected to the Ukrainian Wikipedia. Table 5.1 provides an overview of the general statistics.

Table 5.1: General statistics related to the development of the Ukrainian WordNet basis, including the total number of synsets in the PWN3.1, the number and percentage of synsets linked to Wikidata and the Ukrainian Wikipedia.

	PWN3.1	**Linked to Wikidata**	**Linked to Ukrainian Wiki**
# of synsets	127,020	29,730	21,015
% of synsets	100%	23%	17%

Furthermore, we created a Ukrainian Hypernymy Pairs dataset of noun pairs that express hypernymy relations between words. Table 5.2 presents the number of pairs obtained for each relationship type, their examples, and the total number of pairs acquired. Each lexico-semantic relation is described in detail in Section 2.3. Since the PWN offers a partition of hypernyms and hyponyms with their instances, we keep this split in our dataset. The instance hypernym represents the reflexive type, while an instance hyponym denotes a specific instance of something. For example, the instance hypernym of *Dnipro River* is *river*. We discovered a few data samples where the word on the left is the same as the word on the right, resulting from several WordNet IDs linking to the same Wikidata page. We decided to eliminate such entries from the dataset to enhance its quality. We have made the dataset[1] available for public use through the Hugging Face platform. It can be particularly valuable for the Hypernym Detection task, which involves presenting a model with pairs of words and asking it to determine whether a specific relationship exists between them.

A critical limitation of the approach we employed is that each obtained synset is represented by only one lemma. This is because Wikipedia articles are primarily represented by one word,

[1] https://huggingface.co/datasets/lang-uk/hypernymy_pairs

Table 5.2: Ukrainian Hypernymy Pairs dataset statistics. This table presents the number of word pairs obtained for each type of relationship, including hypernym-hyponym, co-hyponyms, hypernym-instance, and co-instances.

Relation Type	**Example Pair**	**# of Pairs**
Hypernym-Hyponym	водойма, море	6,906
Co-Hyponyms	море, озеро	42,860
Hypernym-Instance	море, Чорне море	2,971
Co-Instances	Чорне море, Азовське море	22,927
Total # of Pairs		**275,664**

and linking is on the synset level. As a result, a crucial next step in developing the Ukrainian WordNet would be to add synonyms to the obtained lemma-synsets. However, it is important to note that this falls outside the scope of this bachelor's thesis. The following section focuses on the Hypernym Discovery task and Instruction-Following LLaMA, which were used to generate candidates to fill the gaps in the created WordNet.

5.2 Ukrainian adaptation to SemEval-2018 Task 9

This work presents a replication of the SemEval-2018 Task 9 benchmark for the Ukrainian language; only a general-purpose setup was considered. The following section presents the data collection process and provides global statistics on obtained datasets.

5.2.1 Data Collection

The data collection process by Camacho-Collados et al. (2018) consisted of five sequential steps. Initially, they compiled the source corpus. Then, they created a vocabulary, collected and selected input terms, extracted gold hypernyms, and filtered and validated them.

The Corpus

For the source corpus in our study, we utilized 31GB UberText 2.0[2] (Chaplynskyi, 2023), which comprises of around 2.5 billion tokens. It is the most extensive publically available corpus of Ukrainian and contains information from many diverse domains. UberText 2.0 consists of five subcorpora, and statistics are summarized in Table 5.3:

1. news obtained from 38 news websites covering national, regional, and industry-specific domains;
2. fiction acquired from two public libraries;
3. social media content — 264 public telegram channels, gathered through the TGSearch project[3];
4. the Ukrainian Wikipedia as of January 2023;
5. decisions of the Supreme Court of Ukraine received upon request for public information.

[2] `https://lang.org.ua/en/ubertext/`
[3] `https://tgsearch.com.ua`

Table 5.3: Statistical information overview of UberText2.0, including its subcorpora, time span, number of sources, texts, and tokens. Source: Chaplynskyi (2023)

Subcorpora	**Time Span**	**# of Sources**	**# of Texts**	**# of Tokens**
News	2000-2023	38	7,208,299	2,172,526,177
Fiction	-	2	23,796	253,321,894
Court	2007-2021	1	111,658	285,252,442
Wikipedia	2004-2023	1	2,819,395	499,603,082
Social	2018-2022	264	885,314	63,472,353
Total	-	-	8,592,389	2,489,454,148

Vocabulary

The vocabulary is a comprehensive list of all possible hypernyms. The vocabulary was created to establish a unified hypernym search space. To construct the vocabulary, we considered all words that appeared at least 5 times in the source corpus. We also processed the data to remove hyperlinks and exclude words that did not contain Cyrillic symbols. Although SemEval-2018 organizers included bi- and trigrams in their approach, we chose to focus solely on unigrams. This was due to the time-consuming nature of generating n-grams for such a large corpus.

Input Terms and Gold Hypernyms

The Hypernym Discovery dataset consists of 2 main parts: input hyponym along with its type and gold hypernyms. The type is either concept (hyponym) or a named entity (instance).

The authors of the original task developed a dataset through a sequential process, beginning with the collection of input terms. The term collection process was conducted using a semi-automatic two-pass procedure. In the first pass, terms were automatically extracted from the source corpus. In the second pass, the authors manually validated and refined the preliminary list of input terms by normalizing each item and removing vague or general terms. The type of the term was also labeled manually. The gold hypernyms were extracted using various taxonomies, such as PWN, Wikidata, MultiWiBi (Flati et al., 2016), and Yago (Suchanek, Kasneci, and Weikum, 2007), via inter-resource mappings provided by BabelNet (Navigli and Ponzetto, 2010). The hypernym extraction process involved retrieving all the synsets that included the given term and then iteratively visiting the father nodes across all the reference taxonomies up to five levels and selecting all the lemmas of the traversed synsets that appeared in the vocabulary file.

We leveraged our WordNet basis to create a similar dataset for the Ukrainian language. The input terms, i.e., hyponyms and corresponding gold hypernyms, were extracted automatically from the obtained resource in a single pass. Both direct and indirect hypernyms up to five nodes in the tree were included, as in the original setup. Obtained input terms and corresponding hypernyms were filtered with several techniques:

- the removal of too broad terms (level 3 at the top of the WordNet graph);
- normalization of entries by removing information from brackets;
- deletion of non-unigrams;
- removal of titles consisting of Latin letters;
- removal of input terms for which no direct hyponym was extracted.

The frequency threshold for all terms to appear at least five times in the source corpus was also preserved. The type of the input terms was obtained automatically using the relation parameter of the synset (instance or hyponym).

Statistics

The dataset statistics are shown in Figure 5.1, which depicts the number of input terms in each dataset categorized by its type and in total. Consistent with the original setup, we evenly divided the training and test sets, 2400 each. Furthermore, a smaller-sized trial set was constructed, containing several dozen samples, which could be utilized as a development set. It is worth noting that in most cases, each term is associated with multiple hypernyms. Therefore, counting all the term-hypernym pairs in the dataset would result in more significant figures. For instance, the test set alone includes 5380 hyponym-hypernym pairs.

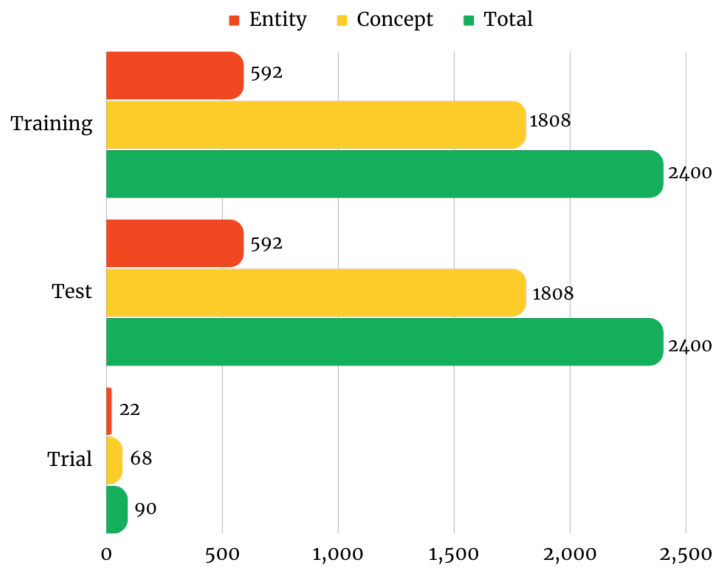

Figure 5.1: Ukrainian Hypernym Discovery dataset statistics, categorized by its type and in total. Training and test sets are split equally; trial refers to the development set.

Overall, our dataset comprises 4890 samples. To provide a point of comparison, the English general-purpose dataset of Camacho-Collados et al. (2018), which was the largest among all the subtasks in SemEval-2018 Task 9, contained 3050 samples. The split was 1500/1500/50 for the training, test, and trial sets.

5.2.2 Experimental Setup

This thesis work utilizes the supervised part of the Hybrid Approach to Hypernym Discovery, as implemented by Bernier-Colborne and Barrière (2018) and available on GitHub[4]. The implementation is written in PyTorch (Paszke et al., 2019), a deep learning framework. We trained the model from scratch on a Linux machine with an NVIDIA GeForce RTX 3090 GPU device.

Before training the Hypernym Discovery, the word embeddings are learned and normalized to unit length. The specific embedding methods used are discussed in the following section. During the training process, dropout is applied to the query projections and embeddings of the query and the candidate hypernym. Gradient clipping and an early stopping strategy are

[4] https://github.com/gbcolborne/hypernym_discovery

employed to regularize the model. Training is terminated if the MAP on the trial set does not increase after 50 continuous epochs. The maximum epoch is set to 1000, although experiments have shown that 35 epochs are sufficient for training.

We use the Adam optimizer (Kingma and Ba, 2014) with $\beta_1 = \beta_2 = 0.9$ and a learning rate of $2e - 4$ for all datasets. All hyperparameters mentioned above, except for patience, were chosen based on the original paper.

To evaluate models, we upgraded the SemEval-2018 Task 9 scorer script to Python3, made minor optimizations, and added the proposed MOC metric.

5.2.3 Comparison of Embeddings

We hypothesize that using better word embeddings would enhance the performance of the Hypernym Discovery model. To test this hypothesis, we conducted experiments to compare the effectiveness of different embedding methods, specifically word2vec and fasttext approaches. All experiments were conducted on the pre-tokenized, lowercased corpus with preserved punctuation.

Baseline

To establish a baseline, we trained 200-dimensional word2vec[5] word embeddings, as was performed in the initial paper. The skipgram architecture was chosen, which uses a target word for predicting the context by summing the log probabilities of the surrounding words to the left and right of the target word. Word2vec vectors were trained with ten negative samples, a window size of 7, and a sample threshold of $1e - 5$. Therefore, the baseline is the Hypernym Discovery model trained with word2vec embeddings.

Fasttext Embeddings

In addition to the baseline, we also analyzed the model's performance using fasttext embeddings (Bojanowski et al., 2017). Fasttext[6] is a popular embedding method that can capture subword information, making it particularly effective for languages with complex morphology like Ukrainian. We trained fasttext using the skipgram algorithm, 2-5 subword size, and 15 negative samples. These parameters were defined as optimal for the Ukrainian language in the previous research (Romanyshyn, Chaplynskyi, and Zakharov, 2023). The vector size was increased to 300, and the rest of the hyperparameters were the same as for word2vec.

It is important to note that word2vec provides a *-read-vocab* parameter that allows passing produced for the task vocabulary file, not constructing it from the training data. In contrast, fasttext does not provide such functionality. Fasttext produces two output files: a `.bin` file with model weights and a plain-text `.vec` file, where a word from the model's vocabulary is assigned its vector. The first option lets us get vectors for out-of-vocabulary (OOV) words. We utilized both ways to obtain the vectors for the task pre-compiled vocabulary in the experiments.

5.2.4 Results

Tables 5.4 summarize the model's performance by the metric and entity type being evaluated. Overall, we can see that the HD_Fasttext_vec model performed the best overall, but HD_Fasttext_bin achieved the highest score in terms of MOC (27.63).

Our results align with those reported by Camacho-Collados et al. (2018), where all models performed better on entities than on concepts. One possible explanation for this trend is that

[5] `https://code.google.com/archive/p/word2vec`
[6] `https://fasttext.cc`

Table 5.4: Our system's performance on the test set for the adapted Ukrainian version of the SemEval 2018 Task 9. HD_Baseline refers to the Hypernym Discovery model with word2vec embeddings, HD_Fasttext_bin to the one using binary fasttext, and HD_Fasttext_vec utilized plain-text fasttext vectors. The best score for each model is marked in **bold**.

	MOC	MRR	MAP	P@1	P@3	P@6
			All			
HD_Baseline	26.55	29.23	20.84	25.25	20.22	19.3
HD_Fasttext_bin	**27.63**	28.7	19.87	22.42	19.53	18.76
HD_Fasttext_vec	25.3	**31.84**	**21.44**	**27.62**	**20.23**	**19.32**
			Concepts			
HD_Baseline	17.84	21.7	12.5	17.48	11.42	11.16
HD_Fasttext_bin	**18.56**	21.02	12.44	14.49	11.94	11.75
HD_Fasttext_vec	17.2	**25.39**	**14.13**	**21.46**	**12.53**	**12.1**
			Entities			
HD_Baseline	53.15	**52.23**	**46.32**	**48.99**	**47.1**	**44.16**
HD_Fasttext_bin	**55.33**	52.14	42.58	46.62	42.71	40.19
HD_Fasttext_vec	50.04	51.54	43.76	46.45	43.75	41.38

supervised systems benefit from the inherent lexical memorization, as entity hypernyms, such as *провінція* and *метрополіс*, occur frequently.

5.3 Instruction-Following LLaMA

To fine-tune LLaMA-7B on hypernymy instructions, we used a parameter-efficient tuning technique called low-rank adaptation (LoRA) (Hu et al., 2021). LoRA reduces the number of trainable parameters for downstream tasks by freezing the pre-trained model's weights and adding trainable rank decomposition matrices into each layer of the transformer architecture (Maurya, 2023). The same GPU device as in Section 5.2.2 was utilized for training the fine-tuned model. The inspiration and fine-tuning hyperparameters were taken from UAlpaca[7].

We constructed the training instructions based on our Hypernym Discovery dataset using different strategies that are presented below.

5.3.1 Instructions Generation

We created instruction datasets of three different types and ran experiments on them. The data for Hypernym Discovery was the basis, and the same split per training/test/trial (dev) was kept for proper comparison with Hypernym Discovery results. The main difference is that we merged training and trial sets into one.

Lean

The most basic way to create an instruction was to directly ask the model to generate as many hypernyms for a given word as there are in gold hypernyms. An example of such an instruction is: *Згенеруй мені п'ять гіперонімів до слова "лаванда".*, where the input term is *лаванда*, and the number of hypernyms is taken from the gold set. Such an approach results in 2490

[7] https://github.com/robinhad/kruk

instructions for the experiment. As shown in Table 5.5, the model with such a limited set of instructions performs poorly. Therefore, we proposed several data augmentation techniques to produce additional training data.

Full

In the full setup, we aimed to create a more diverse and comprehensive set of instructions by designing 19 patterns for each query hyponym. Initially, the patterns were generated by ChatGPT (OpenAI, 2022) and then manually validated to ensure their relevance and diversity. This allowed us to generate a total of 47,310 fine-tuning samples, which is a significant increase compared to the lean setup. Our instruction patterns included a wide range of questions, such as *Які терміни відносяться до вищого рівня абстракції в порівнянні з "лаванда"?* or *Чи є ще загальні категорії, до яких можна віднести "лаванда"?* Each pattern is phrased differently, even if the intended meaning is the same, to ensure that the model can generalize to different types of instructions. With this more extensive and diverse set of instructions, the model's performance improved significantly, as displayed in Table 5.5.

Multiple

In the previous experiment, we fine-tuned the model on a single training data class, i.e., hypernym relation. Further, we hypothesized that the model could benefit from multiple classes of training data, including the opposite relation and co-hyponyms. Therefore, we generated instruction patterns for three relation types: hypernyms, hyponyms, and co-hyponyms. We kept the same 19 patterns for hypernyms as in the full setup. The 13 co-hyponyms patterns were created, such as *Які інші терміни можна використовувати як когіпоніми до "лаванда"?* and *Запропонуй мені когіпоніми до слова "лаванда"*. Hyponyms had 14 patterns, for instance, *Які поняття є більш конкретними, ніж "лаванда?"* or *Запропонуй гіпоніми до слова-гіпероніма "лаванда"*. Such a strategy resulted in a diverse set of 78,149 fine-tuning samples. As we can see from Table 5.5, the model's performance decreased when more relation classes were included. Despite the increased diversity in the training data, the model may have become too generalized and less effective at identifying specific hypernymy relation. Our study indicates that additional exploration is required to establish the most effective trade-off between diversity and specificity of fine-tuning instructions.

5.3.2 Results

We conducted testing of all three models using the same input queries and gold hypernyms from the Hypernym Discovery dataset. However, as the fine-tuning instruction for Lean model is more straightforward than in other experiments, we propose two heuristics to obtain reliable results.

For the lean setup, we tested the model on a single instruction: *Згенеруй мені шість гіперонімів до слова "input_term"*. Nevertheless, we observed that the model sometimes failed to generate any answers or suggested only one word duplicated six times. Therefore, we provided each instruction to the model three times and combined the results while keeping all unique candidates.

In Full and Multiple, we did not encounter this problem with duplicates anymore, but models occasionally failed to generate any candidates when asked to suggest six hypernyms. However, they give some options when you do not limit the number of output hypernyms. Hence, we randomly selected four hypernym instructions and tested each input hyponym against them. The instructions were:

1. Надай мені декілька гіперонімів до слова "input_term".

2. Надай мені шість гіперонімів до слова "input_term".
3. Які слова є гіперонімами поняття "input_term"?
4. Які загальні поняття описують слово "input_term"?

When combining the results, we considered the frequency of the proposed candidates, i.e., the model often proposes the identical hypernym several times. If the resulting candidates had the same frequency, we sorted them in the order the model proposed them. We aimed to make the evaluation process as close as possible to that of Hypernym Discovery. Nevertheless, there were still instances where the resulting predictions were fewer or more than six options. When there were more, we trimmed the values from the tail to calculate the metrics. However, it is important to mention that we kept all unique candidates for the annotator.

The results in Table 5.5 indicate that the LoRA_Hypernymy_Full model outperforms the Hypernym Discovery models in all metrics, with significant improvements observed in all query types. Additionally, the instruction-following LoRA models preserve the ability to predict hypernyms for Entities better than Concepts.

Table 5.5: The LoRA fine-tuning results with hypernymy instructions using different setups. The "LoRA_Hypernymy_Lean" setup only uses the most basic hypernymy instructions, while "LoRA_Hypernymy_Full" includes 19 instruction patterns for a single input query. In the "Multiple" setup, three relation types (hypernym, hyponym, and co-hyponym) were used in addition to diverse patterns.

	MOC	**MRR**	**MAP**	**P@1**	**P@3**	**P@6**
			All			
LoRA_Hypernymy_Lean	6.38	4.54	2.92	3.08	2.88	2.8
LoRA_Hypernymy_Full	**41.61**	**42.6**	**36.74**	**39.0**	**36.27**	**35.93**
LoRA_Hypernymy_Multiple	37.07	35.48	31.19	30.42	31.72	30.8
			Concepts			
LoRA_Hypernymy_Lean	6.16	3.17	2.34	1.71	2.47	2.45
LoRA_Hypernymy_Full	**37.67**	**39.94**	**32.69**	**35.67**	**32.03**	**31.63**
LoRA_Hypernymy_Multiple	33.19	31.92	27.07	25.77	27.46	26.77
			Entities			
LoRA_Hypernymy_Lean	7.06	8.71	4.72	7.26	4.11	3.88
LoRA_Hypernymy_Full	**53.63**	**50.73**	**49.1**	**49.16**	**49.21**	**49.07**
LoRA_Hypernymy_Multiple	48.91	46.35	43.78	44.59	44.71	43.11

5.4 Error Analysis

In addition to quantitative results, we also conducted a qualitative analysis of the outputs of two top-performing models based on MOP scores from our experiments. The metrics comparison of HD_Fasttext_bin and LoRA_Hypernymy_Full on all entity types are presented in Figure 5.2.

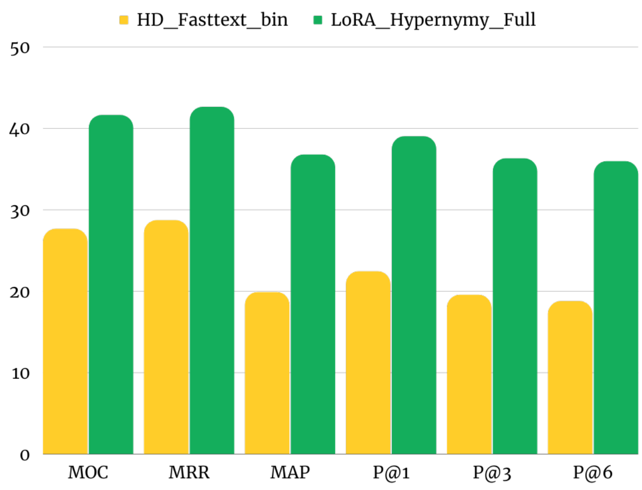

Figure 5.2: Metrics comparison of the two top-performing models based on MOC score for all entity types. The models are HD_Fasttext_bin and LoRA_Hypernymy_Full.

To investigate the models' predictions, we sampled several random examples from our testing dataset. Table 5.6 analyses the HD_Fasttext_bin, which achieved the highest total MOC score (27.63) at the Hypernym Discovery task. The table shows the model's performance on two query types, concepts and entities, including the overlap coefficient (OC), and predicted hypernyms for each query.

Table 5.6: Examples of predictions made by Hypernym Discovery model HD_Fasttext_bin, showing input hyponym (query), its type, overlap coefficient (OC), and top-6 predicted hypernyms for high and low score. The OC of 100 indicates that all ground truth hypernyms were predicted, and 0 means that the model proposed no relevant options. Bolded examples are correct hypernyms.

Query	**Type**	**OC**	**Predictions**
			High OC examples
фалафель	Concept	100	дистрикт, **страва**, **їжа**, метрополіс, муніципалітет, організм
молюски	Concept	67	**організм**, їжа, **безхребетні**, дистрикт, артефакт, рослини
Ориноко	Entity	100	організм, дистрикт, муніципалітет, метрополіс, артефакт, **річка**
Неаполь	Entity	100	**метрополіс**, **дистрикт**, **муніципалітет**, артефакт, організм, **порт**
			Low OC examples
холестерол	Concept	17	їжа, дистрикт, організм, артефакт, **речовина**, муніципалітет
їдальня	Concept	0	їжа, організм, артефакт, метрополіс, дистрикт, будівля
Гіпатія	Entity	17	дистрикт, **організм**, муніципалітет, артефакт, їжа, метрополіс
Сапфо	Entity	0	метрополіс, артефакт, організм, дистрикт, муніципалітет, їжа

We observed that the model tends to overfit to frequently occurring hypernyms such as *метрополіс, муніципалітет, дистрикт*, and *організм*, resulting in incorrect predictions. However, when these candidates are true hypernyms, the model generally ranks them as the top predictions. Moreover, the MOC metrics prove helpful in cases where the ground truth contains only one hypernym, such as the *річка* hypernym for the *Ориноко* query and the model ranks it as the last candidate. Notably, the model can also suggest relevant candidates absent in the ground truth, as observed in the Low OC Entity examples, where it proposed *організм* as a hypernym for *Сапфо*, which is not the direct hypernym but still relevant as it is the same case as for query hyponym *Гіпатія*, where the *організм* was present in gold hypernyms.

Table 5.7 provides prediction examples of the best-performing hypernymy instruction-following LoRA model. The model appears to be confident in its predictions, often providing the same answer for four instructions, resulting in fewer variants of predictions. For instance, it predicts the single hypernym *річка* for the input term *Неккар*. In the LoRA model, the memorization problem of frequent hypernyms is less noticeable. However, the model can still accurately predict such options when they are true hypernyms, such as for *Манчестер*. We also notice the LLaMA's multilingual nature, with some options appearing in English, such as *Retriever*.

Table 5.7: Examples of predictions made by the LoRA_Hypernymy_Full model with their corresponding query types, OC scores, and predicted hypernyms. The examples are divided into two sections: high OC examples, where the model performed well, and low OC examples, where the model's predictions were less accurate.

Query	**Type**	**OC**	**Predictions**
			High OC examples
лабрадор-ретривер	Concept	100	**ретривер**, хижі, тварини, Retriever
холангіт	Concept	100	**симптом, запалення**, хвороба
Неккар	Entity	100	**річка**
Манчестер	Entity	100	**дистрикт, метрополіс, муніципалітет**, порт, столиця
			Low OC examples
метамфетамін	Concept	0	опіати, наркотик, анальгетики
меритократія	Concept	0	диктатура, система, правитель, організм
Сент-Джонс	Entity	0	озеро, річка
Колоси	Entity	0	плід, організм, кущ, рослини

In addition, the model can predict relevant hypernyms that are not present in the ground truth set, such as *хвороба* for the input word *холангіт* and *наркотик* for *метамфетамін*. However, there are instances where the model suggests a co-hyponym instead of a hypernym, as in the case of *диктатура* for *меритократія*.

Another challenge the model faces is the ambiguity of some hyponyms. For instance, by providing the hypernym *річка* for the entity *Сент-Джонс*, the model may have referred to an actual river in Florida, United States, while our data referred to a city in Canada. Similarly, the model's predictions for the entity *Колоси* were relevant only if the hyponym was in the singular form *колос* and of the concept type.

Chapter 6

Conclusions

6.1 Contribution

The research presented in this thesis contributes to the natural language processing field by proposing a data-driven approach for automated hypernym hierarchy construction for the Ukrainian WordNet. The method created a solid foundation for the new WordNet resource by mapping PWN, Wikidata, and Wikipedia. Furthermore, we proposed a simple Gap Ranking algorithm for identifying the best gap nodes for filling.

Different techniques were suggested to generate candidates for filling the gaps: one using the current missing node in the tree and two others using information about its children.

In this work, we adapted SemEval 2018 Task 9: Hypernym Discovery to the Ukrainian language by creating proper datasets and employing an existing large language corpus.

Furthermore, we explored the capabilities of SOTA LLMs for solving the Hypernym Discovery task. To do so, we demonstrated how to construct a sufficiently large set of instructions from an initial small dataset. Likewise, we showed that LLMs could be fine-tuned to build a chatbot-like assistant specializing in a particular hypernym suggestion task.

We developed a simple tool to assist with the manual annotation of obtained candidates for filling gaps in Ukrainian WordNet, which will be further adapted to lexicographers' needs.

This thesis establishes a scalable foundation for creating a comprehensive and reliable WordNet for the Ukrainian language. The future work required for improving and expanding the resource is discussed in Section 6.3. The code for the primary approach[1] and annotation tool[2] are available on GitHub.

6.2 Limitations

Due to time constraints, the scope of this diploma work is limited to establishing the basis of the Ukrainian WordNet by linking existing resources and generating candidates to fill their gaps. Consequently, a significant amount of work refining and presenting the results to the public remains for future steps.

Our approach is currently limited to identifying hypernyms and hyponyms, and further research is necessary to include other lexico-semantic relations. It should be noted that the proposed method can be adapted for other languages but is dependent on the availability of comprehensive Wikipedia data.

Moreover, a notable limitation of our work is that we use mapping of the Ukrainian language to English in the first step of creating a WordNet basis, which may not capture the language nuances and could contain errors. Therefore, further professional verification and input from linguists are necessary.

[1] https://github.com/lang-uk/wikidrill

[2] https://github.com/romanyshyn-natalia/annotation-tool

6.3 Future Work

Since the work on creating WordNet is a long process, there are many open areas for future research. We have proposed an iterative plan where the Hypernym Discovery dataset will be enlarged at every step, and the models will be improved based on input from a professional annotator. However, the finalization of such a pipeline is left for future work.

Our first next steps are as follows:

1. Given that Wikipedia is a resource that is constantly updated, we plan to rerun the linking algorithm of Wikidata and Ukrainian Wikipedia to get more initial pairs. Also, with the help of annotated gaps, we can independently add links to Wikidata, thereby improving this resource.

2. In addition, in this study, the Hypernym Discovery dataset was limited to unigrams; to approximate the setting of SemEval 2018 Task 9, we will also include bigrams and trigrams in the dataset, which in turn will significantly increase its size.

3. Work on adapting SemEval 2018 Task 9: Hypernym Discovery to the Ukrainian language, including datasets, corpus, and trained models, serves as a solid baseline to present this task to the Ukrainian NLP community. Future proposed solutions to the task can be used to further WordNet improvement.

4. Experiments with larger LLaMA models, which can significantly boost performance on our task, are another direction. Furthermore, it would be interesting to experiment with its multilingualism and enrich the dataset for fine-tuning with hypernymy instructions of other languages, for instance, English.

5. An essential next step is to create a high-quality and comprehensive manual for annotators, which will take the WordNet development pipeline to a new level. Also, the annotation tool needs to be hosted on the site and configured for convenient operation.

6. In conclusion, WordNet should have a user-friendly interface accessible to the general public and linked to the OMW.

The end? No! To be continued… (Piasecki, Szpakowicz, and Broda, 2009)

Bibliography

Adeeba, Farah and Sarmad Hussain (2011). "Experiences in Building Urdu WordNet". In: *Proceedings of the 9th Workshop on Asian Language Resources*. Chiang Mai, Thailand: Asian Federation of Natural Language Processing, pp. 31–35. url: https://aclanthology.org/W11-3406.

Anisimov, Anatoly et al. (2013). "Ukrainian WordNet: Creation and Filling". In: *Proceedings of the 10th International Conference on Flexible Query Answering Systems - Volume 8132*. FQAS 2013. Granada, Spain: Springer-Verlag, 649–660. isbn: 9783642407680. doi: 10.1007/978-3-642-40769-7_56. url: https://doi.org/10.1007/978-3-642-40769-7_56.

Basile, Valerio (June 2015). "WordNet as an Ontology for Generation". In: *WebNLG 2015 1st International Workshop on Natural Language Generation from the Semantic Web*. Nancy, France. url: https://hal.inria.fr/hal-01195793.

Batsuren, Khuyagbaatar et al. (July 2019). "Building the Mongolian WordNet". In: *Proceedings of the 10th Global Wordnet Conference*. Wroclaw, Poland: Global Wordnet Association, pp. 238–244. url: https://aclanthology.org/2019.gwc-1.30.

Berend, Gábor, Márton Makrai, and Péter Földiák (June 2018). "300-sparsans at SemEval-2018 Task 9: Hypernymy as interaction of sparse attributes". In: *Proceedings of the 12th International Workshop on Semantic Evaluation*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 928–934. doi: 10.18653/v1/S18-1152. url: https://aclanthology.org/S18-1152.

Bernier-Colborne, Gabriel and Caroline Barrière (June 2018). "CRIM at SemEval-2018 Task 9: A Hybrid Approach to Hypernym Discovery". In: *Proceedings of the 12th International Workshop on Semantic Evaluation*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 725–731. doi: 10.18653/v1/S18-1116. url: https://aclanthology.org/S18-1116.

Bian, Jiang et al. (2008). "Finding the Right Facts in the Crowd: Factoid Question Answering over Social Media". In: *Proceedings of the 17th International Conference on World Wide Web*. WWW '08. New York, NY, USA: Association for Computing Machinery, 467–476. isbn: 9781605580852. doi: 10.1145/1367497.1367561. url: https://doi.org/10.1145/1367497.1367561.

Bird, Steven, Ewan Klein, and Edward Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc."

Bojanowski, Piotr et al. (2017). "Enriching Word Vectors with Subword Information". In: *Transactions of the Association for Computational Linguistics* 5, pp. 135–146. doi: 10.1162/tacl_a_00051. url: https://aclanthology.org/Q17-1010.

Bond, Francis and Ryan Foster (Aug. 2013). "Linking and Extending an Open Multilingual Wordnet". In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics, pp. 1352–1362. url: https://aclanthology.org/P13-1133.

Bond, Francis and Kyonghee Paik (Jan. 2012). "A Survey of WordNets and their Licenses". In: url: https://bond-lab.github.io/pdf/2012-gwc-wn-license.pdf.

Bond, Francis et al. (2016). "CILI: the Collaborative Interlingual Index". In: *Proceedings of the 8th Global WordNet Conference (GWC)*. Bucharest, Romania: Global Wordnet Association, pp. 50–57. url: https://aclanthology.org/2016.gwc-1.9.

Camacho-Collados, José (2017). "Why we have switched from building full-fledged taxonomies to simply detecting hypernymy relations". In: *ArXiv* abs/1703.04178. url: https://arxiv.org/pdf/1703.04178.pdf.

Camacho-Collados, Jose et al. (June 2018). "SemEval-2018 Task 9: Hypernym Discovery". In: *Proceedings of the 12th International Workshop on Semantic Evaluation*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 712–724. doi: 10.18653/v1/S18-1115. url: https://aclanthology.org/S18-1115.

Chaplynskyi, Dmytro (May 2023). "Introducing UberText 2.0: A Corpus of Modern Ukrainian at Scale". In: *Proceedings of the Second Ukrainian Natural Language Processing Workshop*. Dubrovnik, Croatia: Association for Computational Linguistics, pp. 1–10. url: https://aclanthology.org/2023.unlp-1.1.

Commons, Wikimedia (2017). *Hypernym/hyponym (and co-hyponym) relation*. url: https://commons.wikimedia.org/wiki/File:Hyponym_and_hypernym.svg.

Deerwester, Scott C. et al. (1990). "Indexing by Latent Semantic Analysis". In: *J. Am. Soc. Inf. Sci.* 41, pp. 391–407. url: http://wordvec.colorado.edu/papers/Deerwester_1990.pdf.

Devlin, Jacob et al. (June 2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. doi: 10.18653/v1/N19-1423. url: https://aclanthology.org/N19-1423.

Elkateb, Sabri et al. (2006). "Arabic WordNet and the Challenges of Arabic". In: *Proceedings of the International Conference on the Challenge of Arabic for NLP/MT*. London, UK, pp. 15–24. url: https://aclanthology.org/2006.bcs-1.2.

Espinosa-Anke, Luis et al. (Nov. 2016). "Supervised Distributional Hypernym Discovery via Domain Adaptation". In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, pp. 424–435. doi: 10.18653/v1/D16-1041. url: https://aclanthology.org/D16-1041.

Flati, Tiziano et al. (Sept. 2016). "MultiWiBi: The multilingual Wikipedia Bitaxonomy Project". In: *Artificial Intelligence* 241. doi: 10.1016/j.artint.2016.08.004.

Gangemi, Aldo et al. (2003). "Sweetening WORDNET with DOLCE". In: *AI Magazine* 24.3, p. 13. doi: 10.1609/aimag.v24i3.1715. url: https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/1715.

Goodman, Michael Wayne and Francis Bond (Jan. 2021). "Intrinsically Interlingual: The Wn Python Library for Wordnets". In: *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 100–107. url: https://aclanthology.org/2021.gwc-1.12.

Hanna, Michael and David Mareček (Nov. 2021). "Analyzing BERT's Knowledge of Hypernymy via Prompting". In: *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 275–282. doi: 10.18653/v1/2021.blackboxnlp-1.20. url: https://aclanthology.org/2021.blackboxnlp-1.20.

Hearst, Marti A. (1992). "Automatic Acquisition of Hyponyms from Large Text Corpora". In: *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*. url: https://aclanthology.org/C92-2082.

Hoffman, Pablo, Daniel Graña, and Martin Olveyra (2008). *A fast and powerful scraping and web crawling framework*. url: https://scrapy.org/.

Hu, Edward J. et al. (2021). "LoRA: Low-Rank Adaptation of Large Language Models". In: *CoRR* abs/2106.09685. arXiv: 2106.09685. url: https://arxiv.org/abs/2106.09685.

Jing, Hongyan (1998). "Usage of WordNet in Natural Language Generation". In: *Usage of WordNet in Natural Language Processing Systems*. url: https://aclanthology.org/W98-0718.

Johnson, Gregory (2012). "Google Translate http://translate.google.com". In: *Technical Services Quarterly* 29, p. 165. issn: 0731-7131. url: http://www.tandfonline.com/doi/abs/10.1080/07317131.2012.650971.

Kelly, Spencer (2017). *WTF_WIKIPEDIA: A pretty-committed Wikipedia markup parser*. url: https://github.com/spencermountain/wtf_wikipedia.

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980*.

Kulchytsky, I., A. Romaniuk, and Kh. Khariv (2010). "Rozroblennia Wordnet-podibnoho slovnyka ukrainskoi movy [Developing a WordNet-like Dictionary of Ukrainian]". In: *Bulletin of Lviv Polytechnic National University*, pp. 306–318. url: https://ena.lpnu.ua/handle/ntb/6774.

Lan, Yunshi and Jing Jiang (Aug. 2018). "Embedding WordNet Knowledge for Textual Entailment". In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, pp. 270–281. url: https://aclanthology.org/C18-1023.

Levy, Omer et al. (2015). "Do Supervised Distributional Methods Really Learn Lexical Inference Relations?" In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, pp. 970–976. doi: 10.3115/v1/N15-1098. url: https://aclanthology.org/N15-1098.

Liu, Bing (2011). *Web data mining exploring hyperlinks, contents, and usage data*. url: http://dx.doi.org/10.1007/978-3-642-19460-3.

Mao, Rui, Chenghua Lin, and Frank Guerin (July 2018). "Word Embedding and WordNet Based Metaphor Identification and Interpretation". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 1222–1231. doi: 10.18653/v1/P18-1113. url: https://aclanthology.org/P18-1113.

Maurya, Aniket (Apr. 2023). *Accelerating Llama with fabric: A comprehensive guide to training and fine-tuning llama*. url: https://lightning.ai/pages/community/tutorial/accelerating-llama-with-fabric-a-comprehensive-guide-to-training-and-fine-tuning-llama/.

Maziarz, Marek et al. (Dec. 2016). "plWordNet 3.0 – a Comprehensive Lexical-Semantic Resource". In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, pp. 2259–2268. url: https://aclanthology.org/C16-1213.

McCrae, John P. and David Cillessen (Jan. 2021). "Towards a Linking between WordNet and Wikidata". In: *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 252–257. url: https://aclanthology.org/2021.gwc-1.29.

Mikolov, Tomas et al. (2013). *Efficient Estimation of Word Representations in Vector Space*. arXiv: 1301.3781 [cs.CL].

Miller, George A. (1994). “WordNet: A Lexical Database for English”. In: *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*. url: https://aclanthology.org/H94-1111.

Morato, Jorge et al. (2004). “WordNet Applications”. In: pp. 270–278. url: http://www.fi.muni.cz/gwc2004/proc/105.pdf.

Navigli, Roberto and Simone Paolo Ponzetto (July 2010). “BabelNet: Building a Very Large Multilingual Semantic Network”. In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Uppsala, Sweden: Association for Computational Linguistics, pp. 216–225. url: https://aclanthology.org/P10-1023.

Nguyen, Kim Anh et al. (Sept. 2017). “Hierarchical Embeddings for Hypernymy Detection and Directionality”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 233–243. doi: 10.18653/v1/D17-1022. url: https://aclanthology.org/D17-1022.

OpenAI (2022). *Introducing ChatGPT*. url: https://openai.com/blog/chatgpt.

Pannitto, Ludovica, Lavinia Salicchi, and Alessandro Lenci (2017). “AHyDA: Automatic Hypernym Detection with Feature Augmentation”. In: *Italian Conference on Computational Linguistics*. url: https://ceur-ws.org/Vol-2006/paper028.pdf.

Paszke, Adam et al. (2019). “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., pp. 8024–8035. url: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Pianta, Emanuele, Luisa Bentivogli, and Christian Girardi (2002). “MultiWordNet: developing an aligned multilingual database”. In: *Proceedings of the First International Conference on Global WordNet*. url: http://multiwordnet.fbk.eu/paper/MWN-India-published.pdf.

Piasecki, Maciej, Stanisław Szpakowicz, and Bartosz Broda (2009). *A Wordnet from the Ground Up*. Wroclaw: Oficyna Wydawnicza Politechniki Wroclawskiej. url: http://www.plwordnet.pwr.wroc.pl/main/content/files/publications/A_Wordnet_from_the_Ground_Up.pdf.

Prager, John, Dragomir Radev, and Krzysztof Czuba (2001). “Answering What-Is Questions by Virtual Annotation”. In: *Proceedings of the First International Conference on Human Language Technology Research*. url: https://aclanthology.org/H01-1006.

Prud’hommeaux, Eric and Andy Seaborne (2008). *SPARQL Query Language for RDF*. W3C Recommendation. http://www.w3.org/TR/rdf-sparql-query/. url: http://www.w3.org/TR/rdf-sparql-query/.

Qiu, Wei et al. (June 2018). “NLP_HZ at SemEval-2018 Task 9: a Nearest Neighbor Approach”. In: *Proceedings of the 12th International Workshop on Semantic Evaluation*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 909–913. doi: 10.18653/v1/S18-1148. url: https://aclanthology.org/S18-1148.

Roller, Stephen, Douwe Kiela, and Maximilian Nickel (July 2018). “Hearst Patterns Revisited: Automatic Hypernym Detection from Large Text Corpora”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 358–363. doi: 10.18653/v1/P18-2057. url: https://aclanthology.org/P18-2057.

Romanyshyn, Nataliia, Dmytro Chaplynskyi, and Kyrylo Zakharov (May 2023). “Learning Word Embeddings for Ukrainian: A Comparative Study of FastText Hyperparameters”. In: *Proceedings of the Second Ukrainian Natural Language Processing Workshop*. Dubrovnik, Croatia: Association for Computational Linguistics, pp. 20–31. url: https://aclanthology.org/2023.unlp-1.3.

Ronzon, Thomas (2018). "DeepL - überraschend anders". Deutschland. In: *Javaspektrum* 2, p. 69.

Rudnicka, Ewa, Wojciech Witkowski, and Maciej Piasecki (Jan. 2021). "A (Non)-Perfect Match: Mapping plWordNet onto PrincetonWordNet". In: *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 137–146. url: https://aclanthology.org/2021.gwc-1.16.

Santus, Enrico et al. (May 2016). "Nine Features in a Random Forest to Learn Taxonomical Semantic Relations". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. Portorož, Slovenia: European Language Resources Association (ELRA), pp. 4557–4564. url: https://aclanthology.org/L16-1722.

Shwartz, Vered, Enrico Santus, and Dominik Schlechtweg (Apr. 2017). "Hypernyms under Siege: Linguistically-motivated Artillery for Hypernymy Detection". In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Valencia, Spain: Association for Computational Linguistics, pp. 65–75. url: https://aclanthology.org/E17-1007.

Siegel, Melanie and Francis Bond (Jan. 2021). "OdeNet: Compiling a GermanWordNet from other Resources". In: *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 192–198. url: https://aclanthology.org/2021.gwc-1.22.

Siemiński, Andrzej (2011). "WordNet Based Word Sense Disambiguation". In: *Computational Collective Intelligence. Technologies and Applications*. Ed. by Piotr Jędrzejowicz, Ngoc Thanh Nguyen, and Kiem Hoang. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 405–414.

Suchanek, Fabian M., Gjergji Kasneci, and Gerhard Weikum (2007). "Yago: A Core of Semantic Knowledge". In: *Proceedings of the 16th International Conference on World Wide Web*. WWW '07. New York, NY, USA: ACM, pp. 697–706. isbn: 978-1-59593-654-7. doi: 10.1145/1242572.1242667. url: http://doi.acm.org/10.1145/1242572.1242667.

Thoongsup, Sareewan et al. (Aug. 2009). "Thai WordNet Construction". In: *Proceedings of the 7th Workshop on Asian Language Resources (ALR7)*. Suntec, Singapore: Association for Computational Linguistics, pp. 139–144. url: https://aclanthology.org/W09-3420.

Touvron, Hugo et al. (2023). *LLaMA: Open and Efficient Foundation Language Models*. arXiv: 2302.13971 [cs.CL].

Tufis, Dan, D Cristeau, and Sofia Stamou (Jan. 2004). "BalkaNet: Aims, Methods, Results and Perspectives – A General Overview". In: *Romanian Journal of Information Science and Technology Special Issue* 7, pp. 9–43. url: https://profs.info.uaic.ro/~dcristea/papers/tufis-cristea-stamou2004.pdf.

Van Rossum, Guido and Fred L. Drake (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace. isbn: 1441412697.

Vaswani, Ashish et al. (2017). "Attention is All You Need". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 6000–6010. isbn: 9781510860964. url: https://arxiv.org/pdf/1706.03762.pdf.

Vossen, P. (2002). "WordNet, EuroWordNet and Global WordNet". In: *Revue française de linguistique appliquée* VII, p. 27. url: https://www.cairn.info/revue-francaise-de-linguistique-appliquee-2002-1-page-27.htm.

Vossen, Piek, ed. (1998). *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. USA: Kluwer Academic Publishers. isbn: 0792352955. url: https://www.ercim.eu/publication/ws-proceedings/DELOS3/Vossen.pdf.

Vossen, P.J.T.M. (1997). “EuroWordNet: a multilingual database for information retrieval”. English. In: *Proceedings of the DELOS workshop on Cross-language Information Retrieval, March 5-7, 1997 Zurich*. Proceedings of the DELOS workshop on Cross-language Information Retrieval, March 5-7, 1997, Zurich. Vrije Universiteit. url: https://dl.acm.org/doi/10.5555/314515.

Vrandečić, Denny (2013). “The Rise of Wikidata”. In: *IEEE Intelligent Systems* 28.4, 90–95. issn: 1541-1672. doi: 10.1109/MIS.2013.119. url: https://doi.org/10.1109/MIS.2013.119.

Wang, Shan and Francis Bond (Oct. 2013). “Building the Chinese Open Wordnet (COW): Starting from Core Synsets”. In: *Proceedings of the 11th Workshop on Asian Language Resources*. Nagoya, Japan: Asian Federation of Natural Language Processing, pp. 10–18. url: https://aclanthology.org/W13-4302.

Weeds, Julie et al. (Aug. 2014). “Learning to Distinguish Hypernyms and Co-Hyponyms”. In: *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin City University and Association for Computational Linguistics, pp. 2249–2259. url: https://aclanthology.org/C14-1212.

Wei, Tingting et al. (2015). “A semantic approach for text clustering using WordNet and lexical chains”. In: *Expert Systems with Applications* 42.4, pp. 2264–2275. issn: 0957-4174. doi: https://doi.org/10.1016/j.eswa.2014.10.023. url: https://www.sciencedirect.com/science/article/pii/S0957417414006472.

Youn, Hyejin et al. (2016). “On the universal structure of human lexical semantics”. In: *Proceedings of the National Academy of Sciences* 113.7, pp. 1766–1771. doi: 10.1073/pnas.1520752113. url: https://www.pnas.org/doi/abs/10.1073/pnas.1520752113.