Towards the Creation of the Filipino Wordnet: A Two-Way Approach

Briane Paul Samson, Charibeth Cheng, Unisse Chua,
Dan John Velasco, Axel Alba, Trisha Gail Pelagio, Bryce Anthony Ramirez
Robi Jeanne Bangonon, Christine Deticio, Sharmaine Gaw, Danielle Kirsten Sison,
Criscela Ysabelle Racelis, James Kevin Lin, Mark Edward Gonzales, Phoebe Clare Ong
De La Salle University
Manila, Philippines
{briane.samson, charibeth.cheng, unisse.chua,
dan_velasco, axel_alba, trisha_pelagio, bryce_ramirez,
robi_jeanne_bangonon, christine_deticio, sharmaine_gaw, danielle_sison,
criscela_racelis, james_lin, mark_gonzales, phoebs_ong}@dlsu.edu.ph

***Abstract*—As databases of lexical information on words and their lexical relationships, WordNets are important for various downstream natural language processing applications. However, the construction of WordNets can be challenging, especially for low-resource languages such as Filipino. The existing Filipino WordNet has not been maintained, and lacks contextual information for identifying the evolution of word senses. In this study, we built a corpus of 5,370,667 unique tokens and used it to construct a Filipino WordNet via a two-way approach that combines natural language processing and network science. For the natural language processing approach, we utilized only two linguistic sources: our corpus and a RoBERTa-based language model that generates sentence embeddings. For the network science approach, we created a temporal-multiplex network that represents the co-occurrence of words, their semantic relationships, and their usage in different sources across time. We show that our proposed method can induce existing senses (30% of our validation data, as evaluated by matching with the senses from Princeton WordNet) and generate 9,549 semantic sets.**

***Index Terms*—wordnet construction, word sense induction, word sense disambiguation, word co-occurrence networks**

I. Introduction

A language evolves as people make new words, recombine them, or adapt existing ones to make way for emerging ideas. However, lexicologists suggest that people favor reusing existing words and extending their meanings instead, a phenomenon otherwise known as polysemy. A wordnet is a machine-readable database that stores lexical information about a language. It reflects shifts in word senses and aid in understanding semantic relationships. They are used in many tasks and problems in Natural Language Processing (NLP) that require word sense information, such as word sense disambiguation, text classification, text summarization, and machine translation [1]–[4].

The pioneering Princeton WordNet [5] has served as a cornerstone, inspiring the construction of wordnets for different languages such as Filipino. A Filipino WordNet [6] was made by translating from Princeton WordNet into Filipino while also supplementing it with additional words. It contains 14,095 words and 10,188 synsets and was able to document the language's overall structure and a word's morphology, set of synonyms, and part-of-speech. However, this Filipino WordNet has not been maintained and it cannot capture emerging words and senses in Filipino, especially colloquial words prevalent online. For comparison, the second edition of UP Diksiyonaryong Filipino, a Filipino dictionary, contains over 200,000 word senses [7], while the current Filipino WordNet documents only 16,810 word senses. Due to advancements in technology, it is possible to make the Wordnet construction more efficient by making the process more automatic.

In this particular study, a diachronic perspective is employed. In diachronic linguistic studies, two distinct computational approaches allow us to represent and track how word meanings extend and change over time: word embeddings [8]–[11] and word co-occurrence networks [12]. Although both approaches are able to capture linguistic contexts within bodies of text and effectively show historical emergence and changes [11], [13], [14], they still lack information about a word's social context. Previous studies have found that new meanings emerge to minimize cognitive costs [14] and a dominant sense becomes stable through cooperation and competition [11].

However, there is no single sense that becomes dominantly used by the general population. Words evolve to express various, often unrelated meanings, formality and sentiments as they are used by specific populations, cultures and subcultures, age groups and social classes in different fields of expertise and social contexts. In the social sciences, they found new senses emerge in armed conflicts [15], in shifts in the markers of social class [16], and in gender and ethnic stereotypes [17]. Furthermore, polysemy is more prominent now in other media like digital platforms, forums, and social networking sites as the majority of the world's population become digital natives. For example, only on Philippine Twitter is the Filipino word *kalat* used to mean private matters, or inappropriate words or thoughts. This complex dynamism remains an open challenge as we continue to build better language models for intelligent

979-8-3503-3078-6/23/$31.00 ©2023 IEEE

systems applications.

Our study seeks to contribute the following:

- We built the Corpus of Historical Filipino and Philippine English (COHFIE) with 5,370,667 unique tokens, collected from scraping online forums, formal books, social media sites, online encyclopedias, and news sites.
- We propose a method to construct a Filipino WordNet using a two-way approach that combines methods from natural language processing and network science.
- For the natural language processing approach, we fine-tuned the Filipino RoBERTa [18] on COHFIE and NewsPH-NLI [19] to obtain a set of embeddings from which word senses and synsets were induced via unsupervised clustering algorithms.
- For the network science approach, we built a temporal-multiplex network that tracks the co-occurrence of words, their usage in different sources across time, and their semantic relationships.

II. Related Works

A. Sentence Embeddings

Sentence embeddings extend word embeddings by mapping entire sentences to dense vectors. These encode sentences in a vector space, ensuring semantically similar sentences are close in proximity. One method for creating sentence embeddings is to average token embeddings in a sentence [20]. However, this approach ignores word interactions. To address this, [21] modified the BERT model using siamese and triplet network structures to generate meaningful sentence embeddings. By training on SNLI and MultiNLI [22], [23] datasets, they achieved state-of-the-art results in Semantic Textual Similarity (STS) tasks, which focus on measuring text similarity.

B. Word Sense Induction

Existing WSI works use clustering algorithms to represent senses by grouping word usages. The most common algorithm is K-Means [24], However, it needs a predetermined number of clusters which is problematic for WSI due to varying numbers of senses across words. One workaround proposed by [24] is to test different k values and select the one with the highest silhouette score, which measures cluster quality. Another approach used by [25] is Affinity Propagation, which does not require the number of clusters beforehand.

In [26], agglomerative clustering and affinity propagation were employed for WSI in Russian, as these algorithms automatically determine the optimal number of clusters, leading to better results. However, all mentioned works perform clustering only once per word. This paper introduces the 3-STEP clustering approach for WSI, repeating the clustering process three times per word. This method reduces the number of clusters representing the same sense, resulting in smaller yet more diverse sense inventories.

III. Corpus Building

A key component of this study is the building of the Corpus of Historical Filipino and Philippine English (COHFIE); the data sources and token counts are summarized in Table I. Publicly available and legal to publish text data from different sources, namely news sites, books, social media, online forums, and Wikipedia, were pre-processed. Available relevant information depending on the medium such as date published and source were also collected as metadata. Non-lexical tokens, such as emails, emojis, links, and hashtags, were categorized and replaced by special tokens using the regular expressions library[1]. Non-alphanumeric symbols, Question marks (?), exclamation points (!), ellipses (...), periods (.), and line breaks were all treated as sentence delimiters except for hyphens between words. All raw text data were treated as documents wherein each document contains one or more sentences. For example, for news sites, one news article is considered a document, while for social media, specifically Twitter, one tweet is considered one document.

TABLE I
Data Sources and Token Count of Proposed Corpus of Historical Filipino and Philippine English (COHFIE)

Source	Total Num. of Tokens	Mean Sentence Length
Books	8,907,568	23
Wikipedia	9,177,188	32
News Sites	94,466,149	20
Reddit	30,041,725	15
Twitter	214,649,994	11
YouTube	578,128,702	13
PinoyExchange	64,289,785	15
Wattpad	15,433	23
LyricsFreak	87,723	13

Each document was first parsed into sentences, delimited by the aforementioned delimiters, using the sentence tokenizer from NLTK[2]. Each sentence was further parsed into word tokens. Since English and Filipino were both used in the sources, the language of each sentence was first detected through the use of a language detector from the googletrans[3] library. Any sentence classified into a language other than English, Spanish, or Filipino was disregarded. Due to a number of common words between Spanish and Filipino, the language detector can misclassify Filipino sentences as Spanish.

After classifying each sentence, the POS-tagger corresponding to the language detected was used on each sentence to obtain the POS of each word in the sentence. The English POS tagger that was used is from NLTK, and the Filipino POS tagger used is FSPOST from Gramatika[4]. Due to the lack of a Taglish POS-tagger that is publicly available and thoroughly tested by other applications, the monolingual POS-tagger used will simply depend on the language that the language detector specifies for the given text.

IV. Filipino WordNet Construction

The study adopts a diachronic perspective and employs two distinct pipelines: a natural language processing (NLP)

[1]https://docs.python.org/3/library/re.html
[2]https://www.nltk.org/
[3]https://pypi.org/project/googletrans/
[4]https://isipsafe-gramatika.appspot.com

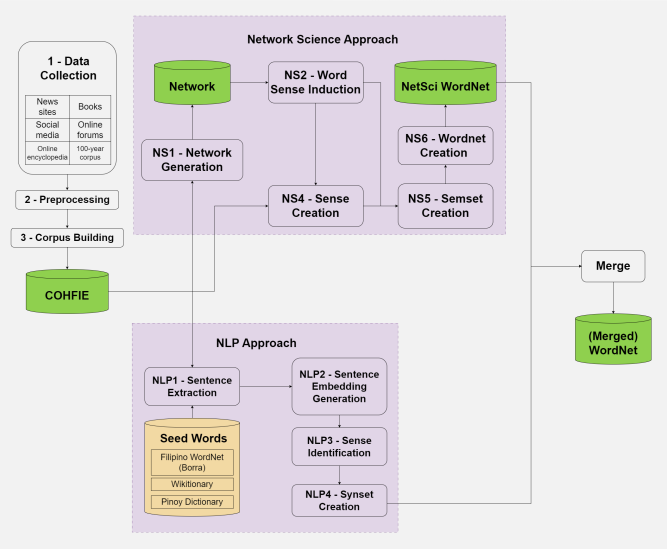

Fig. 1. The pipeline for constructing the Filipino WordNet using a two-way approach.

approach and a network science approach. The process of constructing the Filipino WordNet, or FilWordNet, is illustrated in Figure 1. The subsequent sections provide a detailed discussion of each approach.

A. Natural Language Processing Approach

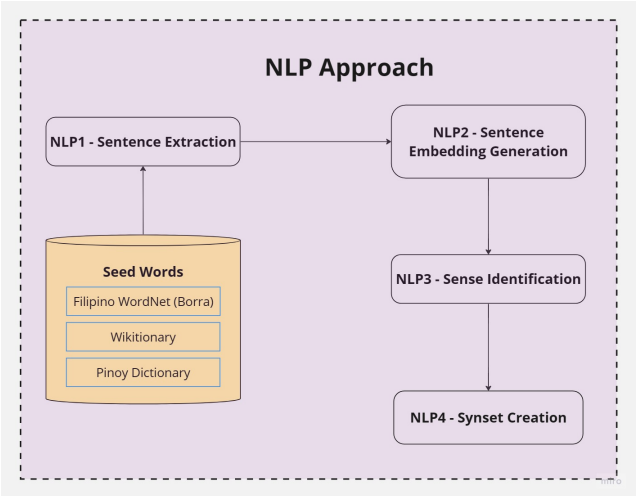

Fig. 2. The natural language processing (NLP) approach pipeline used in creating the Filipino WordNet.

The pipeline for the NLP approach is shown in Figure 2. The following subsections will explain each component.

1) Language Model Training: We finetuned first on masked language modeling to adapt the pretrained model to our corpus. We finetuned the pretrained RoBERTa for Filipino [18] on our corpus for ten (10) epochs or 1,176,690 steps with a maximum learning rate of 5e-5 and then linearly decayed. The language model was optimized with the Adam optimizer [27] using the following hyperparameters: $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e-6$. Masked language modeling experiments were done on Google Compute servers with TPUv3 8 accelerators. The finetuned model is publicly available.

The model is further finetuned on NewsPH-NLI [19], a natural language inference dataset in the Filipino language to generate sentence embeddings. We minimize the Multiple Negatives Rank Loss [28] and only the positive entailment pairs (237,679 sentence pairs) in training. The model was used with Adam optimizer and the following hyperparameters: epochs = 1, learning rate = 2e-5, max sequence length = 128, and batch size = 16. The learning rate is warmed up for the first 1,485 steps or 10% then linearly decayed. Sentence embedding finetuning was done on a personal machine with one NVIDIA GeForce RTX 3060Ti GPU. For the rest of the paper, we will refer to this publicly available model as Sentence-RoBERTa.

2) Sentence Extraction: Initially, sentences from the corpus are extracted. The unique words from the old Filipino WordNet [6] will be used as seed words or target words for the production of senses. These words are used for the query to obtain the example sentences that contain these words from the corpus. A maximum amount is defined, which denotes the number of sentences obtained for each word. For this study, it has been set to 1000 sentences per source due to time and memory constraints. The output for this procedure is the list of texts or sentences that contain the target word specified, along with the following metadata, the year it was published, and the source.

3) Sentence Embedding Generation: After obtaining the sentences upon extraction, the next step is to create sentence embeddings. This allows the data to be represented in such a way that it can determine semantic relations that occur between sentences, which is necessary for Word Sense Induction. The Sentence-RoBERTa was used in order to generate these sentence embeddings. This process is done for each set of example sentences collected from sentence extraction. The output for this module is a set of sentence embeddings for each example sentence that contains the target word.

4) Word Sense Induction: The study utilized WSI techniques like clustering to identify the senses of the words based on the sentence embeddings. WSI clusters sentences that are similar together based on their semantic similarity. Therefore, each cluster is considered a sense. The initial clustering algorithm will produce a lot of irrelevant and redundant clusters. This study then proposed a 3-STEP clustering approach to reduce these clusters into a few yet correct clusters. All three clustering steps would use Affinity Propagation as it does not require the number of clusters to be known in advance. For the first and last clustering steps, PURGING and TRIMMING of clusters were implemented. In PURGE, weak clusters are removed as they have three (3) members or less and are too small to be interpreted by humans. In TRIM, only the N-nearest neighbors or sentences from the cluster's centroid are kept to strengthen the approach of making small-but-correct clusters by eliminating possible noise in each cluster. Cosine similarity is used to choose the nearest neighbors. For all three

clustering steps, the damping parameter of the AP algorithm is set to 0.5. The range of possible values for the damping parameter is 0.5 to 1, exclusive. Generally, the higher the damping, the lower the number of clusters. For example, setting the damping to 0.999 will result in just 1 cluster.

5) Synset Induction: Synsets are essential components of wordnets and to create them, sense embeddings from the previous module can be clustered further using Agglomerative Clustering. A cosine distance threshold of 0.12 is set in such that only the closest senses will be clustered together.

6) Adding New Data to the Wordnet: A language resource like wordnet should be regularly updated since language evolves. To add new data to the wordnet, we generate a new independent sense inventory based only on new data. We will combine this new sense inventory with the existing wordnet through the Append with Threshold method. It works by comparing each sense between the new and old sense inventory with the use of cosine similarity. If a new sense's highest cosine similarity from the senses of the old inventory is equal or lower than the threshold value, denoting that it is different compared to the old senses, then that sense is considered as a new sense which would then be appended into the old sense inventory. This process is repeated until the senses from the new sense inventory are exhausted.

B. Network Science Approach

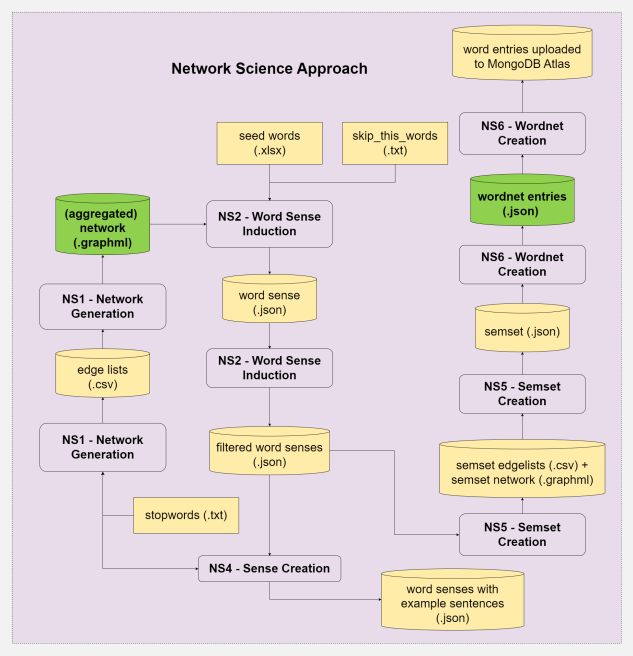

Fig. 3. The network science approach pipeline used in creating the Filipino WordNet.

We illustrate the network science approach pipeline for the Filipino WordNet creation in Figure 3.

1) Network Generation: The corpus underwent preprocessing to address repeated word patterns (>3 instances), remove punctuation, contractions, single-letter tokens, stopwords, and non-alphanumeric characters. The language network was created, where layers represented data domains, and nodes symbolized words. Edges linked words co-occurring within a window size of 3, while edge lists included word ID pairs and co-occurrence weights. The resulting network captured multiplexity and temporality, enabling examination of layers and temporal patterns.

In order to prepare the network for WSI, an aggregated version of the temporal multiplex network, wherein all edges between two words, regardless of when and where two words co-occurred, was created. This allows for community detection across all sources and time, which would capture all possible senses of the word.

2) Word Sense Induction: After the aggregated network was generated, WSI was performed through community detection in order to identify word senses. The process outlined below has 5 steps: ego network construction, node and edge filtering, community detection, community filtering, and validation using a gold standard, closely following the methodology outlined by [29] .

a) Ego Network Construction: Instead of inducing senses from the entire global co-occurrence network, senses were individually induced in sub-networks centered on target words. This local network is called an ego network, wherein the network is centered on a node—in this case the target word, which is called the "ego", while the neighbors of the ego are called the "alters" [30]. These ego networks, comprising nodes centered around a target word (ego) and its neighbors (alters), were created with nodes within one degree of the ego. This approach accelerated Word Sense Induction (WSI) with a smaller scope [29], while permitting overlap in sense communities, even with hard clustering algorithms, due to the localized nature of the communities.

b) Node and Edge Filtering: To filter out irrelevant edges and nodes in the ego networks, the weight of the edges was considered, since words that co-occur more frequently are more likely to be more relevant to each other. Only alters with edges between itself and the ego that have weights above the top 90th percentile were retained.

c) Community Detection and Filtering: Three community detection algorithms (CW, Louvain, Leiden) were used to create sense communities within each ego network. The Leiden algorithm, with modularity and CPM resolutions, used a resolution parameter of 0.5. Since CW is non-deterministic in nature and the partitions in the network were not changing past 6 iterations, the number of iterations was set to 6. Communities under 10% of total alters were removed as overly abstract or infrequently used [31].

d) Validation Using a Gold Standard: To assess sense community quality, a Gold Standard validation involved 23 annotators clustering sentences for 30 test words. Each target word (10 per annotator) had 15 to 25 COHFIE sentences, yielding around 7 to 8 communities per word. A probability matrix formed from annotators' clusters informed an edge list, used to create a network. Sentence pairs with probability scores below 0.4 were deemed unlikely co-occurrences and

disregarded. Edges with 0 weight were removed. Employing the Louvain algorithm, community detection generated distinct senses, which were saved as communities and represented a separate sense of the target word.

e) Validation of Communities: The generated communities from WSI were used to perform WSD for the same sentences that the annotators used for the gold standard. Jaccard index was used to compare the context words of the target words in the sentences with the generated communities from the different WSI algorithms. For each algorithm, the results of WSD were compared against the Gold Standard senses generated using multiple evaluation metrics: NMI, F-measure, and RI. The selected WSI algorithm was the Leiden Modularity algorithm since it had the highest score for most of the metrics used. Additionally, it was the closest to the average community size of the gold standard clusters.

C. Word Sense Disambiguation

WSD was performed in order to get example sentences for each sense, which were stored in the wordnet, using the sense classes produced by WSI and sentences from the corpus. Each input sentence from the corpus was first tokenized and pre-processed. The resulting list of words was then compared to the induced sense communities from performing WSI by computing their similarity score using the Jaccard index. The sense community that resulted in the highest similarity score was used to label the sense of the input sentence. Through this, sense-tagged sentences were obtained, which served as example sentences in the wordnet.

D. Semset Creation

After obtaining the sense representations of target words from WSI, the similarity between each community was measured in order to detect which words share similar sense representations. Afterwards, an edge list was created, wherein each node corresponds to a sense, while the edge corresponds to their similarity, with the weight as the Jaccard similarity score. The edge list was saved as a CSV file. Using the edge list, the network was generated using iGraph and saved as a GraphML file.

Afterwards, edge filtering was performed. Edges with weights less than 0.3 were removed, as they were deemed not similar enough. Afterwards, community detection was then performed using the Leiden algorithm with modularity as the quality function with maximum community size as 10. In order to control and lessen the noise within the semantic set of relationships, the community size parameter was utilized for semset creation. The researchers set the default value of this parameter based on the distribution observed from the size of the synsets from the FilWordNet by Borra et al. [6]. This helped with inspecting the quality of the semsets and preventing extremely large semsets from being generated by the algorithm. The resulting communities were then considered as semsets.

Semantic relationships between senses in semsets were identified. Semsets were manually annotated with different relationships. Additional relationships “related” and “names” were also added.

E. Wordnet Construction

After obtaining the semsets, the entries for the wordnet—called the FilWordNet—was constructed. Besides the word sense, contextual info as well as example sentences from COHFIE obtained from WSD were also stored.

A wordnet entry contains the word the entry represents and a sense and semset ID to identify the word sense. Additionally, a list of example sentences produced in WSD was also stored, as well as contextual information, which pertains to the number of times the word sense is used per source per year from COHFIE.

F. Iteration

To continuously incorporate emerging word senses into the FilWordNet, temporal multiplex network, and wordnet, all processes mentioned in the chapter will be performed iteratively. Data collection is done at different intervals depending on the source, such as daily for Twitter and monthly for most news sites. Since the collection of new data from Wikipedia occurs every two months, which is the longest time between collection for the study, the processes occurring after data collection will be run every two months for new data collected for all sources during that span. This involves pre-processing, adding the new data to the corpus and the temporal multiplex network, performing WSI and WSD, and adding entries to the wordnet, to ensure better efficiency since computation and processes would be performed on more data at once, rather than for only a few new data frequently. Validation for WSI will continuously be done after WSI; however, once the most optimal community detection algorithm and its parameters that can find the best communities have been detected, validation will no longer be done as often. This is similar to that of the validation for WSD. This will lead to a more automated process as it will require less human participation.

V. Future Directions

Due to the limitations of the machine-translated Princeton WordNet as validation data, the presented results account for only 23% of our proposed Filipino WordNet. In this regard, a future direction is building an annotation tool that can be used by linguists and other domain experts to validate the induced senses, synonym sets, and semantic sets.

Automatic methods can introduce errors and inaccuracies, making validation vital to ensure high-quality, accurate synsets. This validation should involve language experts who assess both the automatically generated synsets and the initial Filwordnet. Validation fosters user trust by guaranteeing synset reliability. While our method handles disambiguation, it lacks domain-specific knowledge. Incorrect disambiguation can lead to text misinterpretation. Validation empowers experts to not only check accuracy but also enrich synsets with domain-specific information, often absent in general-purpose resources.

The validation process provides a valuable benchmark for assessing our automatic method's performance, gauging its alignment with human judgments. Additionally, initial inspection reveals that the generated semsets encompass relations like hyponymy/hypernymy, meronymy/holonymy, antonymy, and derivation. Evaluating semsets and their relation labeling can be integrated into the validation process.

VI. Conclusion

In this work, we addressed the construction challenges of WordNets for low-resource languages like Filipino by devising a novel approach that combines natural language processing and network science. Our methodology, centered on a diverse 5-million-token corpus (COHFIE) and a two-way construction strategy, leverages sentence embeddings derived from a fine-tuned Filipino RoBERTa model and a temporal-multiplex network capturing co-occurrence patterns. The results showcase our method's effectiveness, inducing existing senses and producing 9,549 semantic sets, enriching our understanding of semantic evolution. By combining diachronic perspectives with advanced NLP and network analysis, we contribute a robust solution for automating the generation of information needed in the construction of a comprehensive WordNet, which is vital for diverse downstream applications in an ever-evolving linguistic landscape.

Acknowledgment

This research is funded by the Philippine Department of Science and Technology under its Cradle program.

References

[1] A. R. Pal and D. Saha, "An approach to automatic text summarization using wordnet," in *2014 IEEE international advance computing conference (IACC)*, pp. 1169–1173, IEEE, 2014.

[2] K. Bellare, A. D. Sarma, A. D. Sarma, N. Loiwal, V. Mehta, G. Ramakrishnan, and P. Bhattacharyya, "Generic text summarization using wordnet.," in *LREC*, 2004.

[3] S. Scott and S. Matwin, "Text classification using wordnet hypernyms," in *Usage of WordNet in Natural Language Processing Systems*, 1998.

[4] N.-T. Le and N. Pinkwart, "Question generation using wordnet," in *Proceedings of the 22nd International Conference on Computers in Education*, pp. 95–100, 2014.

[5] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, "Introduction to wordnet: An on-line lexical database," *International Journal of Lexicography*, vol. 3, pp. 235–244, 1993.

[6] A. Borra, A. Pease, R. Edita-Roxas, and S. Dita, "Introducing filipino wordnet," in *Principles, Construction and Application of Multilingual Wordners: Proceedings of the 5th Global WordNet Conference*, January 2010.

[7] R. Lim, "One tongue," Aug. 2010.

[8] Y. Kim, Y.-I. Chiu, K. Hanaki, D. Hegde, and S. Petrov, "Temporal analysis of language through neural language models," 2014.

[9] W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Diachronic word embeddings reveal statistical laws of semantic change," 2018.

[10] T. Szymanski, "Temporal word analogies: Identifying lexical replacement with diachronic word embeddings," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, (Vancouver, Canada), pp. 448–453, Association for Computational Linguistics, July 2017.

[11] R. Hu, S. Li, and S. Liang, "Diachronic sense modeling with deep contextualized word embeddings: An ecological view," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 3899–3908, Association for Computational Linguistics, July 2019.

[12] T. C. Silva and D. R. Amancio, "Discriminating word senses with tourist walks in complex networks," *The European Physical Journal B*, vol. 86, jul 2013.

[13] A. Jatowt and K. Duh, "A framework for analyzing semantic change of words across time," in *Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries*, p. 229–238, IEEE Press, 2014.

[14] C. Ramiro, M. Srinivasan, B. Malt, and Y. Xu, "Algorithms in the historical emergence of word senses," *Proceedings of the National Academy of Sciences*, vol. 115, 02 2018.

[15] A. Kutuzov, E. Velldal, and L. Øvrelid, "Temporal dynamics of semantic relations in word embeddings: an application to predicting armed conflict participants," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, (Copenhagen, Denmark), pp. 1824–1829, Association for Computational Linguistics, Sept. 2017.

[16] A. C. Kozlowski, M. Taddy, and J. A. Evans, "The geometry of culture: Analyzing the meanings of class through word embeddings," *American Sociological Review*, vol. 84, pp. 905–949, sep 2019.

[17] N. Garg, L. Schiebinger, D. Jurafsky, and J. Zou, "Word embeddings quantify 100 years of gender and ethnic stereotypes," *Proceedings of the National Academy of Sciences*, vol. 115, apr 2018.

[18] J. C. B. Cruz and C. Cheng, "Improving large-scale language models and resources for filipino," 2021.

[19] J. C. B. Cruz, J. K. Resabal, J. Lin, D. J. Velasco, and C. Cheng, "Exploiting news article structure for automatic corpus generation of entailment datasets," in *PRICAI 2021: Trends in Artificial Intelligence* (D. N. Pham, T. Theeramunkong, G. Governatori, and F. Liu, eds.), (Cham), pp. 86–99, Springer International Publishing, 2021.

[20] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," in *ICLR*, 2017.

[21] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, (Hong Kong, China), pp. 3982–3992, Association for Computational Linguistics, Nov. 2019.

[22] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, (Lisbon, Portugal), pp. 632–642, Association for Computational Linguistics, Sept. 2015.

[23] A. Williams, N. Nangia, and S. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, (New Orleans, Louisiana), pp. 1112–1122, Association for Computational Linguistics, June 2018.

[24] M. Giulianelli, M. Del Tredici, and R. Fernández, "Analysing lexical semantic change with contextualised word representations," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (Online), pp. 3960–3973, Association for Computational Linguistics, July 2020.

[25] M. Martinc, S. Montariol, E. Zosa, and L. Pivovarova, "Capturing evolution in word usage: Just add more clusters?," *CoRR*, vol. abs/2001.06629, 2020.

[26] N. Arefyev, B. Sheludko, and T. Aleksashina, "Combining neural language models for word sense induction," *ArXiv*, vol. abs/2006.13200, 2020.

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[28] M. Henderson, R. Al-Rfou, B. Strope, Y. hsuan Sung, L. Lukacs, R. Guo, S. Kumar, B. Miklos, and R. Kurzweil, "Efficient natural language response suggestion for smart reply," 2017.

[29] M. Bekavac and J. Šnajder, "Graph-based induction of word senses in Croatian," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, (Portorož, Slovenia), pp. 3014–3018, European Language Resources Association (ELRA), May 2016.

[30] A. Kaveh, M. Magnani, and C. Rohner, "Defining and measuring probabilistic ego networks," *Social Network Analysis and Mining*, vol. 11, 12 2021.

[31] D. Jurgens, "Word sense induction by community detection," in *Proceedings of TextGraphs-6: Graph-based Methods for Natural Language Processing*, (Portland, Oregon), pp. 24–28, Association for Computational Linguistics, June 2011.