Automatic Identification of Filipino Metaphors using Machine Learning

Lorenzo Miguel L. Ocampo and Maria Art Antonette D. Clariño

***Index Terms*— Filipino, Metaphor, BERT, RoBERTa, Detection**

I. Introduction

A metaphor is a type of figure of speech that is used because of its ability to bring depth to the meanings of messages [1]. It is typically used to represent other meanings using other words rather than the literal word itself. For example, the sentence ”Siya ay isang ahas”, the word ”ahas” does not literally mean that the person is an animal. It is an expression that means that the person exhibits characteristics of the animal ”ahas”. Therefore, the meaning of the word ”ahas” in this sentence differs from its literal definition. Since metaphors are an important part of written and spoken language, it is necessary to understand the use of these expressions in the field of natural language processing. There has also been a growing interest in the automatic identification of metaphors in the field of NLP since it can benefit other tasks in this line of study [2]. Since the standard procedure for using NLP is working with the language in its literal sense like in information retrieval and text classification [3], There have been far less situations where it is being used with figurative language as a model [4].

There are studies that created projects that focused on processing different individual figures of speech. [5] created a method to generate similes to match a given literal sentence. while [6] created a model for generating a metaphoric sequence from a literal expression. On the other hand, [7] created a model that can generate similes, metaphors, idioms, hyperboles, and sarcasm. There are also studies that investigated these figures of speeches in different languages as well. [8] designed and developed a computational tool to analyze the linguistic elements in Hindi poetry. while [9] proposed a different method of identification for Chinese figures of speech. Also, [10] made a cross-lingual metaphor detector for English and Spanish.

However, there have been no investigations whose purpose has been to explore the detection of figures of speech or figurative language in the Filipino language . Therefore, this work intends to design a program that is capable of automatically detecting Filipino figures of speech.

A. Objectives

Overall, the study aims to design an algorithm that can detect Filipino metaphors and modify the sentence embedding model. The specific objectives of the study are the following:

1) Build a Corpus containing Filipino sentences and whether they have metaphors in the sentence;
2) Pre-train and fine-tune the Filipino-sentence-roberta-v1 model sentence embedder using the corpus built;
3) Develop a Filipino metaphor detection algorithm that utilizes both metaphor identification procedure and selectional preference violation; and
4) Measure the performance of the detection algorithm using precision, recall and F1 scores.

B. Significance of the study

The detection of metaphors in Filipino language can aid not only other tasks in NLP but also future researches of similar fields [11] [12]. This study will be useful in the field of machine translation, particularly detecting the segments of a text that cannot be directly translated since they carry metaphorical expressions of different meanings [2]. Also by determining whether or not there is a presence of metaphors in a text, this study will be helpful in the job of sentiment analysis since analyzer will know whether the text must be interpreted in its literal meaning or not [13].

C. Scope and limitations

The study will only involve the detection of metaphors written in the Filipino language. The data set that will be used will be taken from credible online news articles within the year of 2022 and some articles from political discourse.

II. Review of Related Literature

A. Machine Learning

Machine learning is an application of Artificial Intelligence or AI that allows systems to learn and improve from experience without being explicitly programmed. It focuses more on the development of computer programs that can use data to learn for themselves [14]. It achieves this by training the system's algorithm to learn based on a given set of data either by supervised or unsupervised learning. Supervised machine learning uses training data where the outputs are already known and trains the algorithm by optimizing the model parameters until it can produce an output similar to the training data. Unsupervised learning is where the training data does not have a predetermined correct output and uses the algorithm to examine the information and identify a structure from within the data. Some examples of where machine learning is used are self-driving cars [15], cyber fraud detection [16], and online recommendation engines from Facebook, Netflix, and

Presented to the Faculty of the Institute of Computer Science, University of the Philippines Los Baños in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science

Amazon. Machine learning allows these by filtering useful information and mixing them together to produce these results [17]. There are many fields under machine learning that focus on replicating how humans think, one example of this is NLP or natural language processing.

B. Natural Language Processing

NLP is a branch of AI that uses computational techniques to analyze and represent language to achieve human-like language processing for a large range of applications and tasks [18]. Since NLP aims to achieve human-like understanding of language, some of its goals include paraphrasing text, translating between languages, answering questions from the contents of the text, and drawing inferences from the text [19]. Some applications of NLP include smart assistants like Apple's Siri and Amazon's Alexa, predictive text features like autocorrect and autocomplete, email filters that filter spam emails, and language translation [20]. Although NLP is typically used for literal language, it can also be applied to figurative language.

C. Figurative Language

Figurative language is a type of literary device that adds color to language [21]. It is used to express oneself without the use of a word's real meaning. Commonly found in comparisons and exaggerations, it is used to add a creative element to written or spoken language or to explain complicated ideas. [22]. It has a fundamental impact on the readers. By linking concepts, images, or objects that have no relation with each other by creating new connections between them, readers can discover new insights and see a clearer or imaginative image in their heads. It can transform what readers deem to be ordinary into something significant [23]. A common way to express figurative language is through the use of different figures of speech.

D. Figures of Speech

A figure of speech is a word or phrase that means something different than what it seems to say, as opposed to the literal meaning of the expression [24]. It is consciously used to be more artistic yet subtle, thus making them have more intellectual and emotional impact, more memorableness, and more depth compared to the literal meanings of the expressions [25]. There are several types of figures of speech being used. These include similes, metaphors, hyperboles, personification, synecdoche, and onomatopoeia. While figurative language refers to the language that uses or contains figures of speech, figures of speech are the particular techniques being used. If figurative language can be compared to a dance routine, then the figures of speech are the different moves that makes up the routine [26]. A challenge being faced when using NLP with figures of speech is their detection in a body of text.

E. Detection

Detection is the act of noticing something that is partially hidden or unclear, or to discover something using a special method [27]. It can also be used to refer to the process of extraction of particular information from a much larger collection of information without the cooperation from or synchronization with the sender [28]. It came from the word ”detector” which was first used to refer to a device that detected the simple presence or absence of a radio signal, since all radio signals were sent in Morse code. The term is still being used today to describe a unit that extracts particular signals from all electromagnetic waves present [29].

F. Related Studies

Figures of speech and figurative language are commonly used in the English language [3]. However, there are many kinds of figures of speech and they are also different across different countries and languages. Several studies have already used NLP to process these figures of speech in English and as well as their own languages. The application of NLP to figures of speech is usually divided into three different tasks: the detection of figurative language, the interpretation or identification of its literal meaning, and the generation of new figures of speech. This research will focus on figures of speech detection.

There have been many previous studies that have worked with different individual figures of speech. This study by [5] worked on the automatic generation of similes. Another study by [6] also implemented an automatic generator but targeting metaphors this time. Aside from those studies, there are also other studies that target multiple figures of speech at the same time. This study by [7] implemented the detection and generation of similes, metaphors, idioms, hyperbole, and sarcasm.

Other than focusing on different types of figurative languages, there are also studies that chose to concentrate on different languages instead. In this study by [8] , the aim was to develop a computational tool to process figures of speech in Hindu poetry. Another study by [9] built a Chinese corpus for Contextualized Figure Recognition that performs figure extraction, figure type classification, and figure recognition for Chinese figures of speech. Other studies like [10] performed metaphor detection for these figures of speech in the Spanish language instead.

There were many methods that were used by different researches, specifically in the task of metaphor detection. there are studies like [30] that use recurrent neural networks and [31] that use graph convolutional neural networks to detect word metaphoricity . However, neural network based solutions have limitations in representing certain context of words [32]. To counter this problem, some other studies use contextualization-based approaches to detect metaphors like BERT (Bidirectional Encoder Representations from Transformers) [33] and RoBERTa (A Robustly Optimized BERT Pretraining Approach) [34]. This researched created a metaphor detector called MIss RoBERTa WiLDe that uses the RoBERTa masked language model with wiktionary lexical definitions. On the other hand, this study used a BERT based approach to detect verb metaphors [35].

There are researches that also used different metaphor identification theories to aid their metaphor detection. The most commonly used theories are Selectional Preference Violation (SPV) refers to detecting metaphors based on its semantic difference from its surrounding words [36] and Metaphor Identification Procedure (MIP) refers to detecting metaphors based on how different the literal meaning of the word is compared to the contextual meaning [37]. In this study, the SPV metaphor identification theory was used to detect novel metaphors and it was found that it outperforms an all-metaphor baseline classification [38]. While this study created models for both SPV and MIP where the SPV model uses Recurrent Neural Network Multi-Head Contextual Attention (RNN_MHCA) and the MIP model uses Recurrent Neural Network Hidden GloVe (RNN_HG). On the other hand, some studies combined both SPV and MIP in the models that they created [39]. This study created a metaphor detection via contextualized late interaction using metaphorical identification theories (MelBERT) that combined both SPV and MIP in their model [32].

III. Materials and Methods

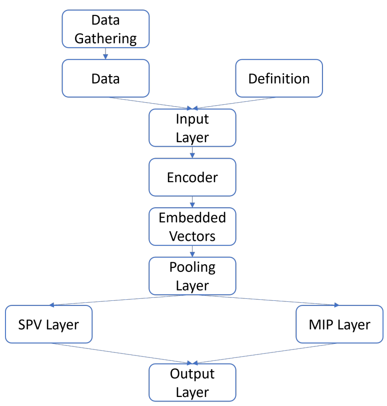

Fig. 1. Model Architecture

The model architecture is inspired by the MelBERT [32] and MIss RoBERTa WIlde [40] because the accuracy of their detection showed significant increase compared to other models. While this work is inspired by these previous works, some changes were made to this research to fit the setting of the study more appropriately. Specifically, the target language is now in Filipino rather than English, and a new type of sentence embedding, that will be fine tuned in this study, will be used rather than the one used in the previous studies.

A. Technology

The programming language that will be used in this study will be python. This is because the model architecture will use python frameworks like HuggingFace, pandas and sentence-transformers to implement the metaphor detector. HuggingFace is a platform used in data science that provides tools that enable developers to build, train, and deploy ML models that are based of open-source technologies. Specifically, the model that will be used is the filipino-sentence-roberta-v1. Pandas is a python library that is used to create data sets. Sentence-transformers is a python framework that allows users to download state-of-the-art sentence, text, and image embeddings from different sources.

B. Data gathering

The data will be used to train the model to detect figures of speech in the input.It will come from various sources of online news posts, articles, or political discourses since these most likely contain metaphors; thus, it will only be comprised of sentences or phrases in the Filipino language, more specifically Tagalog. This will be used for the training, fine-tuning, and evaluation of the application since the model will be programmed to only run for Filipino words. The data collected must have samples of text with and without figures of speech contained in the text. The text will be annotated with the index of the target word and whether the target word is a metaphor or not. The data will also be pre-processed to remove non-alphanumeric characters.

C. Pre Training

The filipino-sentence-roberta-v1 model will be used as the embedding algorithm. The language model is based on RoBERTa for Filipino [41], and it has been pretrained and finetuned using the Corpus of Historical Filipino and Philippine English (COHFIE) [42], and then further finetuned using NewsPH-NLI [43]. While the sentence embeddings were generated from the sentence-RoBERTa model, a model that was able to produce semantically meaningful sentence embeddings through the use of siamese and triplet network structures [44]. It will be further finetuned through the Pytorch trainer API to match the current task and data set that will be used. The data will be split into a proportion of 60% for training, 20% for validation, and 20% for testing since this is split usually used in machine learning projects [45]. Furthermore, the validation data will be divided using K-folds cross validation so that the model can be trained and tested over the complete collection of data. Since it has been shown that it produces fewer errors from very high bias and variance, the value k=10 will be used [46]. Scikit-learn library will be used to divide the data.

D. Definition Layer

The definition layer will contain a list of the definition of the target word . To get the definition of the target word, the target word will undergo lemmatization using FilWordNet [47] to obtain its lexical dictionary form then retrieve its definition using a dictionary scrapper. If the definition cannot be found using the dictionary scrapper, then the lexical dictionary form of the word will be used instead.

E. Input Layer

The input layer will accepts 2 parameters. The sentence data containing the target word taken from the gathered data and the definition of the target word that comes from the definition layer.

F. Encoder

The encoder will accept 2 parameters, the sentence that contains the target word and the definition of the target word then output sentence embeddings for the sentence, the target word, and the definition of the target word. This will be done using a sentence embedder called Filipino Sentence Roberta V1 model. It takes all the words in each input, tokenizes each word in the input, and assigns a vector of 768 hidden values to each input. This vector will be used to compute the values the will be needed in the next layers [42].

G. Pooling Layer

The pooling layer will accept the embedded vectors of the sentence and the definition of the target word. This layer will compute the means of both vectors, concatenate it to the vector representation of the target word, then sends them to the SPV and MIP layer.

H. SPV and MIP Layer

Both of these layers will use cosine similarity to compute for the values since it is used to measure the distance of features of an object [48].

$$\frac{x.y}{||x|| * ||y||}$$

The SPV layer computes the semantic gap between the target word and its surrounding words so it takes the mean of the sentence embeddings and the target word vector. While the MIP layer, that computes the difference between the target word's literal meaning and its contextual meaning, takes the mean of the definition embeddings and the target word vector.

I. Output

The output layer will use the logarithmic softmax function since it is used to convert vectors into probability distributions.

$$\log\left(\frac{exp(x_i)}{\sum_j exp(x_j)}\right)$$

This layer will input the outputs of both the SPV and the MIP layer into the logarithmic softmax function then get the maximum value. The value yields a higher number will be the output of the model.

J. Evaluation

The algorithm will be run using both the original sentence embedder and the fine-tuned sentence embedder and will have its performance evaluated by the precision, recall and F1 performance metrics. Precision will be used to measure the proportion of predicted positive values that are correctly real positives and will be taken with the formula [49].

$$\frac{TP}{TP+FP}$$

Meanwhile, recall will be used to measure the proportion of real positive cases that are correctly predicted positive and will be taken with the formula [49].

$$\frac{TP}{TP+FN}$$

Finally, the F1 score will be used to measure the average of precision and recall and will be taken with the formula [50].

$$\frac{2xPrecisionxRecall}{Precision+Recall}$$

The results will then be compared to each other to check for an improvement in the result after fine-tuning the sentence embedder.

References

[1] A. Isenberg, "On defining metaphor," *The Journal of Philosophy*, vol. 60, no. 21, pp. 609–622, 1963.

[2] R. Mao, C. Lin, and F. Guerin, "Word embedding and wordnet based metaphor identification and interpretation," in *Proceedings of the 56th annual meeting of the association for computational linguistics*. Association for Computational Linguistics (ACL), 2018.

[3] T. Chakrabarty, Y. Choi, and V. Shwartz, "It's not rocket science: Interpreting figurative language in narratives," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 589–606, 2022.

[4] E. Liu, C. Cui, K. Zheng, and G. Neubig, "Testing the ability of language models to interpret figurative language," *arXiv preprint arXiv:2204.12632*, 2022.

[5] T. Chakrabarty, S. Muresan, and N. Peng, "Generating similes effortlessly like a pro: A style transfer approach for simile generation," *arXiv preprint arXiv:2009.08942*, 2020.

[6] T. Chakrabarty, X. Zhang, S. Muresan, and N. Peng, "Mermaid: Metaphor generation with symbolism and discriminative decoding," *arXiv preprint arXiv:2103.06779*, 2021.

[7] H. Lai and M. Nissim, "Multi-figurative language generation," *arXiv preprint arXiv:2209.01835*, 2022.

[8] K. Naaz and N. K. Singh, "Design and development of computational tools for analyzing elements of hindi poetry," *IEEE Access*, vol. 10, pp. 97 733–97 747, 2022.

[9] D. Zhu, Q. Zhan, Z. Zhou, Y. Song, J. Zhang, and S. Li, "Configure: Exploring discourse-level chinese figures of speech," *arXiv preprint arXiv:2209.07678*, 2022.

[10] E. Sanchez-Bayona and R. Agerri, "Leveraging a new spanish corpus for multilingual and crosslingual metaphor detection," *arXiv preprint arXiv:2210.10358*, 2022.

[11] F. R. Lapitan, R. T. Batista-Navarro, and E. Albacea, "Crowdsourcing-based annotation of emotions in filipino and english tweets," in *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016)*, 2016, pp. 74–82.

[12] J. L. Fernandez and K. M. M. Adlaon, "Exploring word alignment towards an efficient sentence aligner for filipino and cebuano languages," in *Proceedings of the Fifth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2022)*, 2022, pp. 99–106.

[13] J. Liu, N. O'Hara, A. Rubin, R. Draelos, and C. Rudin, "Metaphor detection using contextual word embeddings from transformers," in *Proceedings of the Second Workshop on Figurative Language Processing*, 2020, pp. 250–255.

[14] A. L. Samuel, "Machine learning," *The Technology Review*, vol. 62, no. 1, pp. 42–45, 1959.

[15] Q. Rao and J. Frtunikj, "Deep learning for self-driving cars: Chances and challenges," in *Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems*, 2018, pp. 35–38.

[16] O. S. Yee, S. Sagadevan, and N. Malim, "Credit card fraud detection using machine learning as data mining technique," *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 10, no. 1-4, pp. 23–27, 2018.

[17] A. Papadimitriou, P. Symeonidis, and Y. Manolopoulos, "A generalized taxonomy of explanations styles for traditional and social recommender systems," *Data Mining and Knowledge Discovery*, vol. 24, no. 3, pp. 555–583, 2012.

[18] K. Chowdhary, "Natural language processing," *Fundamentals of artificial intelligence*, pp. 603–649, 2020.

[19] E. D. Liddy, "Natural language processing," 2001.

[20] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: State of the art, current trends and challenges," *Multimedia Tools and Applications*, pp. 1–32, 2022.

[21] J. Bets, "Basic types of literary devices." [Online]. Available: https://examples.yourdictionary.com/basic-types-of-literary-devices.html

[22] M. Ellis, "Figurative language examples: How to use these 5 common types," Jul 2022. [Online]. Available: https://www.grammarly.com/blog/figurative-language/

[23] J. Sitbon, "What is figurative language? definition and examples," Jul 2022. [Online]. Available: https://www.wix.com/wordsmatter/blog/2020/10/figurative-language/

[24] A. F. Rosa and P. A. Eschholz, *Models for writers: Short essays for composition*. St. Martin's Press, 1998.

[25] "Figure of speech." [Online]. Available: https://www.britannica.com/art/figure-of-speech

[26] "Figurative language." [Online]. Available: https://www.litcharts.com/literary-devices-and-terms/figurative-language

[27] "Detect." [Online]. Available: https://dictionary.cambridge.org/us/dictionary/english/detect

[28] S. Qian, "What is detection?" Apr 2014. [Online]. Available: https://www.scirp.org/journal/paperinformation.aspx?paperid=44601

[29] "Detection task." [Online]. Available: http://www.gabormelli.com/RKB/Detection_Task

[30] A. Mykowiecka, A. Wawer, and M. Marciniak, "Detecting figurative word occurrences using recurrent neural networks," in *Proceedings of the Workshop on Figurative Language Processing*, 2018, pp. 124–127.

[31] D. Le, M. Thai, and T. Nguyen, "Multi-task learning for metaphor detection with graph convolutional neural networks and word sense disambiguation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 8139–8146.

[32] M. Choi, S. Lee, E. Choi, H. Park, J. Lee, D. Lee, and J. Lee, "Melbert: metaphor detection via contextualized late interaction using metaphorical identification theories," *arXiv preprint arXiv:2104.13615*, 2021.

[33] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[34] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[35] W. Song, S. Zhou, R. Fu, T. Liu, and L. Liu, "Verb metaphor detection via contextual relation learning," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021, pp. 4240–4251.

[36] Y. Wilks, "A preferential, pattern-seeking, semantics for natural language inference," *Artificial intelligence*, vol. 6, no. 1, pp. 53–74, 1975.

[37] P. Group, "Mip: A method for identifying metaphorically used words in discourse," *Metaphor and symbol*, vol. 22, no. 1, pp. 1–39, 2007.

[38] H. Haagsma and J. Bjerva, "Detecting novel metaphor using selectional preference information," in *Proceedings of the Fourth Workshop on Metaphor in NLP*, 2016, pp. 10–17.

[39] R. Mao, C. Lin, and F. Guerin, "End-to-end sequential metaphor identification inspired by linguistic theories," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 3888–3898. [Online]. Available: https://aclanthology.org/P19-1378

[40] M. Babieno, M. Takeshita, D. Radisavljevic, R. Rzepka, and K. Araki, "Miss roberta wilde: Metaphor identification using masked language model with wiktionary lexical definitions," *Applied Sciences*, vol. 12, no. 4, p. 2081, 2022.

[41] J. C. B. Cruz and C. Cheng, "Improving large-scale language models and resources for filipino," *arXiv preprint arXiv:2111.06053*, 2021.

[42] D. J. Velasco, A. Alba, T. G. Pelagio, B. A. Ramirez, J. C. B. Cruz, and C. Cheng, "Automatic wordnet construction using word sense induction through sentence embeddings," *arXiv preprint arXiv:2204.03251*, 2022.

[43] 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 *Pacific Rim International Conference on Artificial Intelligence*. Springer, 2021, pp. 86–99.

[44] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[45] H. Yoon, "Finding unexpected test accuracy by cross validation in machine learning," *International Journal of Computer Science & Network Security*, vol. 21, no. 12spc, pp. 549–555, 2021.

[46] B. G. Marcot and A. M. Hanea, "What is an optimal value of k in k-fold cross-validation in discrete bayesian network analysis?" *Computational Statistics*, vol. 36, no. 3, pp. 2009–2031, jun 2020. [Online]. Available: https://doi.org/10.1007%2Fs00180-020-00999-9

[47] A. Borra, A. Pease, R. Roxas, and S. Dita, "Introducing filipino wordnet," in *Principles, Construction and Application of Multilingual Wordnets: Proceedings of the 5th Global WordNet Conference*, 2010, pp. 306–310.

[48] P. Xia, L. Zhang, and F. Li, "Learning similarity with cosine similarity ensemble," *Information Sciences*, vol. 307, pp. 39–52, 2015.

[49] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *arXiv preprint arXiv:2010.16061*, 2020.

[50] R. M. AlZoman and M. J. Alenazi, "A comparative study of traffic classification techniques for smart city networks," *Sensors*, vol. 21, no. 14, p. 4677, 2021.