Classification of Polysemous and Homograph Word Usages using Semi-Supervised Learning

Sangjun Han[1] Brian Kenji Iwana[2] Satoru Uchida[3]
[1]Department of Electrical Engineering and Computer Science, Kyushu University
[2]Department of Advanced Information Technology, Kyushu University
[3]Faculty of Languages and Cultures, Kyushu University
{sangjun.han, brian}@human.ait.kyushu-u.ac.jp uchida@flc.kyushu-u.ac.jp

Abstract

Words can have different meanings based on the context of how they are used. Therefore, to help language understanding, it is important to be able to distinguish between the word usages. To do this, we propose to automatically classify word meaning using a Transformer neural network. However, annotating large amounts of word usages for effective machine learning can be time-consuming and expensive. Thus, we propose using unlabeled data for Pseudo Labeling to improve the robustness of the model.

1 Introduction

Words can have different meanings, or *usages*, depending on the context. For example, the word “see” can have many different meanings, such as, to see something with your eyes, to understand something, to imagine something in a particular way, etc. Polysemous words are words with related origins and homographs are words with different origins.

The study of polysemous words and homographs is important for language learning [1, 2]. Language learners often have difficulties knowing the intended meaning of words when using dictionaries [3]. Due to this, it is common for language learners to only use the top definition in dictionaries [4]. Thus, correct identification of word usage can be an important tool.

In order to address this problem, we propose the use of machine learning, namely text classification, to predict the usage of words. Namely, we use a Bidirectional Encoder Representations from Transformer (BERT) [5] neural network to embed words into a word-wise semantic vector and use a classifier to learn the usage based on the vector. Through this, we show that it is possible to predict the word usage within the context of a sentence.

However, one issue with using neural networks is the requirement for data. Specifically, a large amount of annotated data is required to train accurate models and acquiring the annotations can be a time-consuming and expensive process. Therefore, we propose the use of Semi-Supervised Learning (SSL) to make up for the lack of data. SSL combines the use of supervised data, i.e. labeled data, and unsupervised data, i.e. unlabeled data. Specifically, Pseudo Labeling [6] is used. In Pseudo Labeling, the unlabeled data is classified and given Pseudo Labels based on the classifier confidence.

The contributions are as follows:

- We develop a word usage classifier that is able to learn the usage of a word based on the context.
- We demonstrate that SSL, specifically Pseudo Labeling, can help make up for the difficulty of annotating data.
- We show that the word embeddings learned by BERT contain contextual usage information.
- A case study is conducted on specific words with polysemous and homograph definitions to show the effectiveness of our approach.

2 Related Work

While word embedding research [7] and text classification research [8, 9] are widely studied fields, specific word usage classification is not often explored.

In a related problem set, homograph disambiguation aims to differentiate homographs in text, most often used with text-to-speech generation [10, 11]. Notably, SSL has been used for homograph disambiguation in text-to-speech generation Mandarin [12] and Persian [13]. While these methods are similar, they differ in that they only separate homographs by sound and not specifically by meaning. This means that the labels typically follow part of speech, very broad meanings, or labels that have different meanings but the same pronunciations [10]. The aim of our work is to predict the usage of words from a comprehensive set of definitions.

3 Bidirectional Encoder Representations from Transformers (BERT)

Transformers [14] are feed forward neural networks that consist of blocks of a Multi-Head Self-Attention (MHSA) layer and a fully-connected layer. The MHSA layer uses parallel self-attention layers to learn pairwise relationships between tokens of the input. After the MHSA, there is a fully-connected layer. The output of a Transformer layer is vector embeddings corresponding to each input token.

Bidirectional Encoder Representations from Transformer (BERT) [5] is an extension to the original Transformer. Some of the improvements include using a bidirectional self-attention, adding segment encodings, and training using BooksCorpus [15], which contains the contents of 11,038 books.

3.1 Tokenization

The input representations of BERT are constructed of summing token embeddings, positional encodings, and segment encodings [5]. The token embeddings are WordPiece vectors [16] embedded in vectors using a linear layer. The WordPiece vectors represent pieces of words, including full words, in a 30,522 word-part token vocabulary. The positional encodings represent the position that the word piece appears in the sentence, and the segment encoding is the sentence number.

3.2 Embeddings

BERT is trained in an encoder-decoder structure. The encoder layers create an embedding corresponding to each input token vector. This allows Transformer layers to be stacked and to be used with a decoder Transformer for training. Thus, the output of the encoder layers is a set of token-wise vector embeddings. Due to this, as shown in Fig. 1, we use these word embeddings as representations for our classifier.

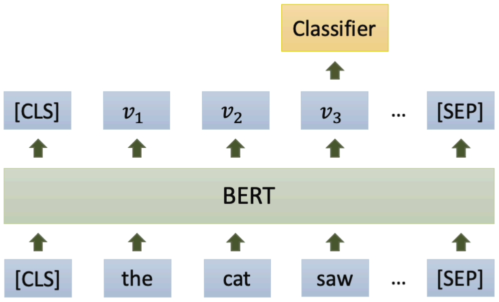

Figure 1 Word Usage Classification Using BERT

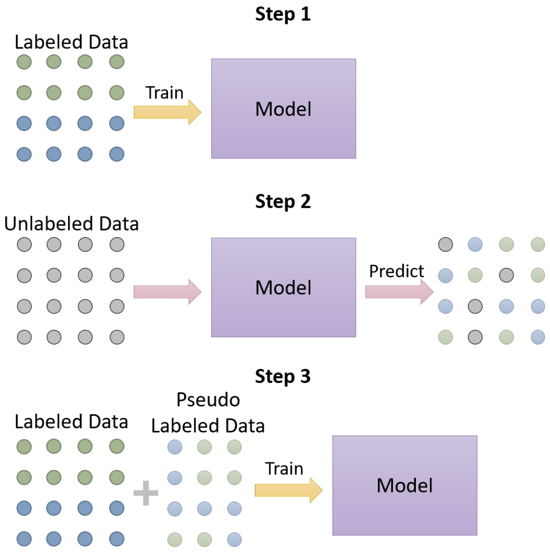

Figure 2 Pseudo Labeling

Aside from the word embeddings, there is a special classification embedding (CLS) and sentence separation embeddings (SEP). In traditional text classification, the CLS token is used to classify the entire document. However, in our case, we do not use the CLS embedding because we focus on individual word usage.

4 Semi-Supervised Learning

SSL is a problem setup that incorporates supervised data and unsupervised data to train models. The benefit of SSL is that it makes it possible to leverage large amounts of unlabeled data to supplement the labeled data. This is useful because annotating unlabeled data can be costly and time-consuming.

One method of SSL is the use of Pseudo Labeling [6]. Pseudo Labeling is a method of labeling the unlabeled data based on probabilistic confidence to be used alongside the already labeled data. Pseudo Labeling is performed in multiple rounds of training, as shown in Fig. 2. First, the model is trained using the labeled data. Next, the unlabeled data is classified using the trained model. When the unlabeled

Table 1 Details of the datasets

Data	Tenses	# Classs	# Train	# Test	# Unlabeled
get	get, got, gotten, gets, getting	70	95	64	703
let	let, lets, letting	16	27	15	84
play	play, played, plays, playing	12	19	12	212
see (1)	see, saw, seen, sees, seeing	20	36	18	550
see (2)	see, saw, seen, sees, seeing	8	894	219	550

data is classified, a confidence score p is determined. For the purpose of Pseudo Labeling, the confidence is defined as the probability of the predicted class. Then, the predicted unlabeled data with a high confidence threshold τ is labeled, i.e. where $p \geq \tau$, which becomes pseudo labels. The model is then trained again using the combined annotated labels and pseudo labels. Finally, this process is repeated N number of rounds or until satisfied.

5 Experimental Results

5.1 Dataset

In this study, we use data from two corpora. The first corpus is the English Vocabulary Profile (EVP) Online word list [17]. The corpus includes a list of words with definitions, usages, and example sentences, hereafter referred to as *dictionary examples*. For this study, the American English definitions are used. In addition, EVP also includes *learner examples* which are sentences written by varying levels of language learners. The second is a privately gathered book corpus consisting of Common European Framework of Reference (CEFR) graded materials. The EVP corpus is used to create the supervised datasets and the book corpus is used for the unsupervised data.

From the corpora, we created five datasets, as shown in Table 1. Each dataset is used to classify the usage of a single word. Also, it should be noted that each dataset incorporates all tenses of the word. Four of the datasets, “get,” “let,” “play,” and “see (1)” use the usage labels determined by EVP. We use the dictionary examples as the training set and the learner examples as the test set. For each unlabeled set, lines of text that contained the specific word were gathered from the book corpus.

In addition to datasets determined by EVP Online, a second dataset, “see (2)” was annotated manually. This dataset combines all sentences with the word “see” (and its tenses) from the entire EVP corpus. Each sentence was annotated based on the eight high-level definitions by two English speakers. The classes are “see (use eyes),” “see (meet),” “see (on media),” “see (understand),” “see (information),” “see (consider),” “see (happen),” and “see (believe).” The problem with the previous datasets is that there are only a few dictionary examples for each class. This dataset is used to evaluate the proposed method but on a larger dataset. The total number of sentences was 1,113 and a training and test split was created by taking 20% from each class to be saved for the test data.

5.2 Architecture and Training

To acquire the word embeddings, a 12 transformer layer pre-trained BERT is used. The pre-trained BERT is fed word piece sequences created from each sentence and outputs word embeddings. As recommended by Bert-as-service [18], the embeddings from the second-to-last transformer layer is used, i.e. the 11th layer. This is done because Bert-as-service found that the last layer embeddings tend to be learned in a way for the Masked Language Model (MLM) [5]. The second-to-last layer contains more contextual information and word meaning.

The word embedding vectors are then classified using a Multi-Layer Perceptron (MLP) neural network. The network consists of two layers, one hidden layer and one output layer. The hidden layer has 512 nodes. The hidden layer uses Rectified Linear Unit (ReLU) activations and Dropout with a probability of 0.5. The weights were initialized using a Xavier uniform initialization [19]. The word embedding vectors are then classified using a fully-connected layer with the number of nodes equalling the number of classes and a softmax activation.

The MLP is trained using Adam optimizer [20] for 1,000 epochs. We use a batch size of 10 and an initial learning rate of 0.0001. For the SSL, rounds of Pseudo Labeling were performed until all of the unlabeled data was labeled or a maximum of 10 rounds has passed. The threshold was set to $\tau = 0.99$.

5.3 Results

The results are shown in Table 2. In the table, MLP is the classifier using the BERT word embeddings without Pseudo Labeling, and MLP+PL uses Pseudo Labeling. For comparison, we use 1-Nearest Neighbor (1-NN) with the

Table 2 Classification Accuracy (%)

Data	1-NN	MLP	MLP+PL
get	50.0	**56.3**	54.7
let	**80.0**	**80.0**	73.3
play	75.0	**83.3**	66.7*
see (1)	**88.9**	77.8	83.3
see (2)	87.2	88.1	**90.0**

* τ = 0.95 instead of τ = 0.99

Table 3 Example Test Set Instances with the Same Token Embedding but Different Word Embeddings That Were Correctly Classified

Prediction	Sentence
see (use eyes)	If I **see** some nice underwear, I will buy it too.
see (consider)	Some people **see** society as it stands today as inherently flawed, an amorphous group of people who follow and worship anyone that gives them pleasure and empty dreams of perfection.
see (meet)	You should **see** a doctor about that cough.

BERT embeddings. For "play," τ = 0.95 was used because no unlabeled data had a confidence of 0.99. The results show that the Pseudo Labeling is able to help improve the accuracy of "see (2)."

Conversely, the datasets with a very high unlabeled to labeled ratio performed worse. Calculated from Table 1, the "get," "let," "play," and "see (1)" datasets have an unlabeled to labeled training data ratio of 7.4:1, 3.1:1, 11.2:1, 15.3:1, respectively. This indicates that Pseudo Labeling is weak in instances where there are not enough training samples compared to the unlabeled samples.

Table 3 shows instances where the target word, "see," had the same input embedding but different output word embeddings. Importantly, the word embedding was able to be used to correctly classify the usage in each sentence. Thus, it can be inferred that the embeddings from BERT contain semantic information and is able to separate homographs.

5.4 Ablation

In order to determine the threshold for Pseudo Labeling, we performed a parameter search. The analysis is performed on the larger "see (2)" dataset to increase the reliability of the analysis. The results in Table 4 show that the best τ is 0.99 for Pseudo Labeling.

5.5 Examining the Pseudo Labels

It is important for the unlabeled data to be assigned accurate Pseudo Labels for SSL to work. Therefore, in Table 5, we examine some of the Pseudo Labels assigned by the model. In the examples where the confidence p was higher than τ = 0.99, the correct Pseudo Label was assigned.

Table 4 Affect of the Threshold Accuracy (%)

PL	τ = 0.90	τ = 0.95	τ = 0.99	τ = 1.00
see (2)	89.0	89.5	**90.0**	89.5

Table 5 Example Pseudo Labels and Confidences for "See (2)"

	Class	p	Sentence
(1)	see (meet)	1.0	A young woman wants to **see** you, sir.
(2)	see (use eyes)	1.0	The view—being up very high looking down and **seeing** the northern lights below.
(3)	see (consider)	1.0	You may feel that other people **see** you as a leader.
(4)	see (on media)	0.93	All of these reach their peak in the bluefin-"the king of all fish, as Ernest Hemingway described them after **seeing** Atlantic bluefin off the coast of Spain.
(5)	see (consider)	0.92	The importance of peer influence can be **seen** clearly in how strongly teenagers react when they fall out with a friend or are excluded from a social peer group.
(6)	see (use eyes)	0.97	But, as you can **see**, the wheels still aren't invisible.

Furthermore, in the table, (4) and (5) were mislabeled. (4) should be "see (use eyes)" and (5) should be "see (understand)." Due to their low confidence scores, they were correctly not Pseudo Labeled. However, despite example (6) having low confidence, it was labeled correctly, thus, not selected as a Pseudo Label.

6 Conclusion

In this research, we verified whether the output word vectors of BERT can represent the usage of words, and made classifiers using the output word vectors. Also, we improved our classifier through the use of Pseudo Labeling. We demonstrate that the use of Pseudo Labeling is useful in helping improve the model. However, there are limitations to Pseudo Labeling and when there are many more unlabeled patterns than there are labeled patterns, then the accuracy is degraded. In the future, we will increase the words and incorporate other information inherent to text to improve Pseudo Labeling.

Acknowledgments

This work was supported by MEXT-Japan (Grant No. JP21K17808, JP22H00677, JP20H00095).

References

[1] Marjolijn Verspoor and Wander Lowie. Making sense of polysemous words. **Language Learning**, Vol. 53, No. 3, pp. 547–586, jul 2003.

[2] Victoria Abou-Khalil, Samar Helou, Brendan Flanagan, Mei-Rong Alice Chen, and Hiroaki Ogata. Learning isolated polysemous words: identifying the intended meaning of language learners in informal ubiquitous language learning environments. **Smart Learning Environments**, Vol. 6, No. 1, nov 2019.

[3] Alex Boulton and Sylvie De Cock. Dictionaries as aids for language learning. In **International Handbook of Modern Lexis and Lexicography**, pp. 1–17. Springer, dec 2016.

[4] Li Jin and Elizabeth Deifell. Foreign language learners' use and perception of online dictionaries: A survey study. **Journal of Online Learning and Teaching**, Vol. 9, No. 4, p. 515, 2013.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.

[6] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In **ICML Workshops**, Vol. 3, p. 2, 2013.

[7] S. Selva Birunda and R. Kanniga Devi. A review on word embedding techniques for text classification. In **Innovative Data Communication Technologies and Application**, pp. 267–281. Springer, 2021.

[8] Andrea Gasparetto, Matteo Marcuzzo, Alessandro Zangari, and Andrea Albarelli. A survey on text classification algorithms: From text to predictions. **Information**, Vol. 13, No. 2, p. 83, feb 2022.

[9] Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S. Yu, and Lifang He. A survey on text classification: From traditional to deep learning. **ACM Transactions on Intelligent Systems and Technology**, Vol. 13, No. 2, pp. 1–41, apr 2022.

[10] Kyle Gorman, Gleb Mazovetskiy, and Vitaly Nikolaev. Improving homograph disambiguation with supervised machine learning. In **Language Resources and Evaluation**, 2018.

[11] Marco Nicolis and Viacheslav Klimkov. Homograph disambiguation with contextual word embeddings for TTS systems. In **ISCA Speech Synthesis Workshop**, 2021.

[12] Binbin Shen, Zhiyong Wu, Yongxin Wang, and Lianhong Cai. Combining active and semi-supervised learning for homograph disambiguation in mandarin text-to-speech synthesis. In **Interspeech**, 2011.

[13] Noushin Riahi and Fatemeh Sedghi. A semi-supervised method for persian homograph disambiguation. In **Iranian Conference on Electrical Engineering**, 2012.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In **Advances in Neural Information Processing Systems (NeurIPS)**, 2017.

[15] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In **International Conference on Computer Vision (ICCV)**, 2015.

[16] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. **arXiv preprint arXiv:1609.08144**, 2016.

[17] English Vocabulary Profile. English vocabulary profile online - american english. https://www.englishprofile.org/american-english, 2023.

[18] Han Xiao. Bert-as-service. https://github.com/hanxiao/bert-as-service, 2018.

[19] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In **International Conference on Artificial Intelligence and Statistics**, pp. 249–256, 2010.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2014.