Ethiopian Journal of Science and Sustainable Development

e-ISSN 2663-3205 Volume 8 (1), 2021

Journal Home Page: www.ejssd.astu.edu.et

Research Paper

Corpus-Based Word Sense Disambiguation for Ge’ez Language

Amlakie Aschale Alemu[1,*], Kinde Anlay Fante[2]

[1]Department of Electrical and computer Engineering, Faculty of Technology, Debre Tabor University, Debre Tabor, Ethiopia
[2]Faculty of Electrical and Computer Engineering, Jimma Institute of Technology, Jimma University, Jimma, Ethiopia

Article Info

Article History:
Received 20 August 2020
Received in revised form 08 December 2020
Accepted 24 December 2020

Keywords:
Natural Language processing
Word Sense Disambiguation
Semi¬-supervised
ADtree
Ge’ez Language

Abstract

In natural language processing, languages have a number of ambiguous words. The absence of automatic word sense disambiguation for any language can be a challenge for the development of natural language processing applications such as Information Extraction, Information Retrieval, Machine Translation, etc. The aim of this study is to design a word sense disambiguation prototype model for Ge’ez Language words using Corpus-based techniques. Due to the unavailability of Ge’ez wordNet and annotated datasets, six ambiguous words were chosen for this study. These words are ሀለፈ (halafe), ቆመ (ḱome), ባረከ (bareke), አስተርዓየ (astaraye), ገብረ (gebre), ሰዓለ (se’ale). A total of 2119 Ge’ez sense examples were collected for the six ambiguous word from Ge’ez literature. The performance of three Corpus-based machine learning techniques (Adaboost, SMO, and ADTree) were tested on the WEKA package. We evaluated the performance of the three Corpus-based machine learning approaches which are unsupervised, supervised and semi-supervised for disambiguation of the six Ge’ez words. The experimental results show that the best performance is achieved using ADtree algorithm (semi-supervised machine learning approach). The proposed method achieved an average performance of 92.1%, 91.3%, 91% and 91.1% of Precision, Recall, F1-score and Accuracy using ADTree algorithm respectively. A window size of 4-4 has been found to be the optimal window size to identify the meaning of the selected ambiguous words of Ge’ez language using ADTree algorithm.

1. Introduction

In the 21st century, the growth of information technology has led the way for a large volume of information to be available for the society. Discussion about importance of a language for using the available information is not far from obvious since it serves as a medium of communication among the races. Language has a potential to express a wide range of ideas and to convey complex thoughts. In particular, natural language is now being used to exchange information among humans and has now reached to the extent of being evolution criteria for the technology. In order to make available information useful for the society, an interest has emerged to make use of technology to process natural language. In response to such a need, Natural Language Processing (NLP) has come up with a main focus of natural language computations (Getahun Wassie and Million Meshesha, 2014).

NLP is a field of computer science that deals with the interaction among computers and humans using natural language that aims to enhance human-to-human communication and human to computer communication (Solomon Mekonnen, 2010). It is normally used to describe the function of software or hardware

[*] Corresponding author, e-mail: amlakieaschale19@gmail.com

https://doi.org/10.20372/ejssdastu:v8.i1.2021.283

components in a computer system, which analyzes or synthesizes spoken or written language. There are in fact two distinct focuses of NLP: language processing and language generation. The former one refers to the analysis of language for producing a meaningful representation, while the second refers to the production of language from a representation (Pal et al., 2013*)*. The field of NLP was originally referred to as Natural Language Understanding (NLU) in the early days of Artificial Intelligence. It is well agreed today that while the goal of NLP is true NLU, that goal has not yet been accomplished. A full NLU System would be able to paraphrase an input text, translate the text into another language, answer questions about the contents of the text and draw inferences from the text (Mahmoodvand and Hourali, 2017*)*. The aim of NLP is studying problems in the automatic generation and understanding of natural languages. Natural language is understood as a tool that people use to express themselves and has specific properties that improves the efficiency of textual information retrieval systems. These properties are linguistic variation and ambiguity. NLP is also a subfield of artificial intelligence and linguistics (Naseer and Hussain, 2009).

Ambiguity is one of the greatest challenges in NLP, the term refers to understanding of something in two or more possible ways or something that has more than one meaning. It can appear in sentence (called structural or syntactic ambiguity) or at a word level (called lexical ambiguity) and phonological ambiguity. Ambiguity is a universally recognized linguistic phenomenon, which arises from the structure of the language and can be explained in terms of the analysis at different levels (Daniel Jurafsky, 2018). So that developing word sense disambiguation (WSD) is crucial for the development of natural language applications such as information extraction, machine translation, information retrieval, question answering, text summarization and others.

In the field of computational linguistics, word sense disambiguation is defined as the problem of computationally determining which “sense” of a word is activated by the use of the word in a particular context. Lexical disambiguation in its broadest definition is nothing less than determining the meaning of every word in context, which appears to be a largely unconscious process in people. Due to the importance of WSD for understanding semantics and many real- world applications, researchers have been interestingly trying to tackle that problem.

So far, different word sense ambiguation techniques were proposed for Amharic (Getahun and Million, 2014; Seid and Yaregal, 2017; Solomon Mekonnen, 2010), Afan Oromo (Workneh Tesema et al., 2016), and Tigrigna (Mersa Mebrhatu, 2018) as languages of Ethiopia. To the best of our knowledge, there is no word sense disambiguation model reported for Ge’ez language. The objective of this work was to develop a word sense disambiguation model for six ambiguous Ge’ez words. We have designed three different corpus-based machine learning models to compare the performance of different techniques. Through experiment, we have explored the best model and the parameters of the models for six ambiguous Ge’ez words.

2. Materials and Methods

This section describes the design of WSD system for Ge’ez language. It mainly focuses on preparation of corpus, word selection, architecture of Ge’ez WSD model, document pre- processing techniques, preparing machine readable datasets, and evaluation techniques of the model. According to different scholars, words that we want to disambiguate could be selected by the researchers from WordNet, which is available on the web or online, or from different sources of the language or documents of the language, which is annotated manually.

2.1. WordNet

WordNet is a lexical database; it provides a large repository of some languages lexical items, which is available online. The WordNet was designed to establish relations between the main four types of Parts of Speech (POS): noun, verb, adjective and adverb. WordNet defines the relations between synsets and relations between word senses. A specific meaning of one word under one type of POS is called a sense and synset represents the smallest unit in WordNet, which describes a specific meaning of a word. It includes the word itself, explanation and the synonyms of its meaning. The difference is that lexical relations are relations between members of two different synsets, however semantic relations are relations between two whole synsets (Workneh Tesema et al., 2016).

2.2. Words that are selected from documents

According to Mersa Mebrhatu, (2018), the construction of the sense tagged corpus needs a great amount of time and cost. Due to this reason, we have selected small a number of ambiguous words in this study. The corpora which have sense information of all words, have been built recently, but they are not large enough to provide sufficient disambiguation information of the all words. Therefore, the methods based on the sense tagged corpora have difficulties in disambiguating senses of all words so that the selection of ambiguous words that were used in this study was based on the number of senses of single word. In Leykun Berhanu (2005), there are words that have multiple senses, from two sense up to sixteen sense. Due to the unavailability of Ge'ez wordNet and annotated datasets, six ambiguous words that have two senses were chosen for this study. These ambiguous words are selected from ግስ (gis) which is found in ቅኔ (kinie) school in Ethiopian Orthodox Tewahido church. WSD performance can be affected by the distribution of training data for each sense that means number of sense examples are required to be equal as much as possible and a balanced distribution of training data has been employed to maximize performance in the work of (Solomon Mekonnen, 2010).

2.3. Data Collection

In this research, we have used corpus-based approach. It is challenging to acquire sense annotated corpus for WSD studies due to lack of standard sense annotated corpus or context-based repository (Wordnets) for Ge'ez language. Due to this reason, data collection becomes first rather than corpus preparation. By considering this, the researchers collected data from different sources such as Ge'ez bible, Sinksar, Fithanegest, Gedile semaetat, and teaching materials from Bahir Dar university Ge'ez Department. Here, we first collected a huge data that contains 193,000 sentences or instances. To retrieve the sentences that contain the selected ambiguous words; we developed a simple algorithm and this simple algorithm accepts a word which is an ambiguous words from the user and then displays the sentences that contain the given ambiguous word and we have got 2,119 sentences or instances from a huge data that we collected. According to Yemane Kaleta et al. (2016), selecting sentences of ambiguous words from a variety of domains is very important to build efficient and reliable WSD prototype since similar domains usually restrict words to one sense. Therefore, in order to build efficient and reliable WSD prototype for Ge'ez language, we collected data from different domain areas.

2.4. Proposed System Architecture

The flow of activities that are used to develop the proposed WSD is given in Figure 1.

The proposed system architecture contains different steps. The first step is accepting sentences that contains ambiguous words. The next step is applying preprocessing activities like normalization, tokenization, stop word removal, stemming and transliteration. In unsupervised learning, unlabeld datasets are given to the selected clustering algorithms to build WSD prototype model of Ge'ez language. Whereas, in supervised learning labeled datasets are given to the selected classification algorithms to build WSD prototype model of Ge'ez language. In semisupervised learning, a few numbers of labeled seed examples together with large number of unlabeled datatasets are given to the selected clustering algorithms inorder to obtain fully labeled datasets based on labeled seed examples. Those fully labeled datasets are given to the selected classification algorithms to build WSD prototype model of the language.

2.4.1. Preprocessing Phase

Preprocessing describes any type of processing performed on raw data to prepare it for next processing procedure. Hence, preprocessing is the preliminary step which transforms the data into a format that will be more easily and effectively processed. Preprocessing must ensure that the source text be presented to NLP is in a form usable for it. In this study, preprocessing is a primary step to make our data sets compatible with the machine learning tool that was used in our study called Weka. In the preprocessing stage of this study tokenization, stemming, stop word removal, transliteration and normalization are performed.

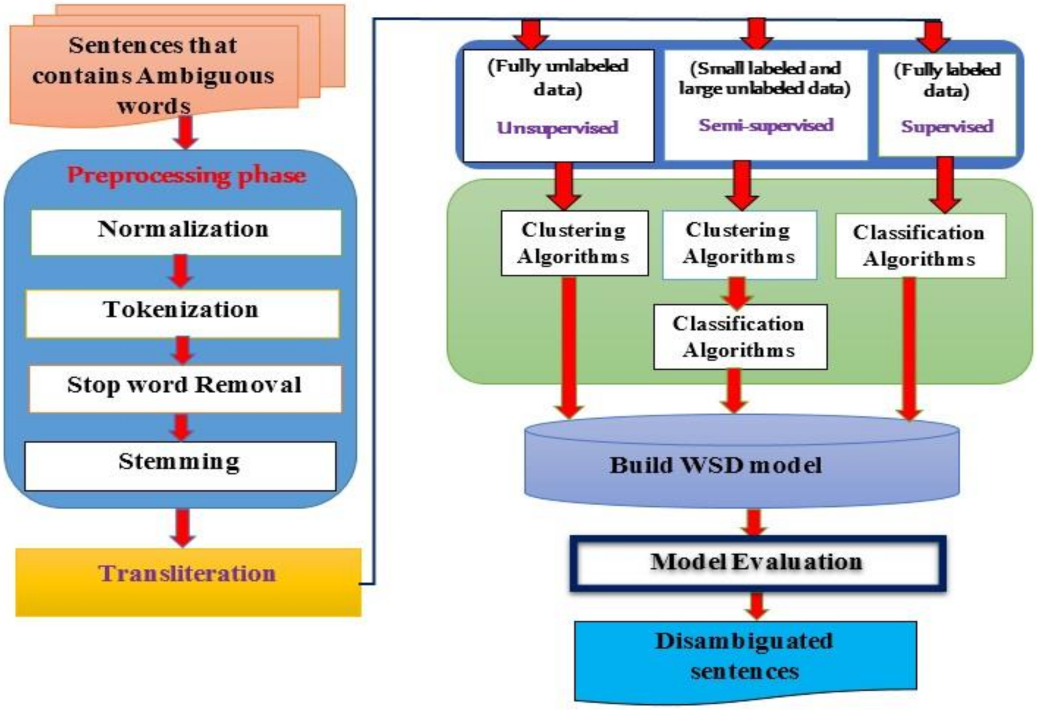


Figure 1: Corpus-Based Ge’ez WSD system Architecture

2.4.1.1. Normalization**:** In this study, normalizing the characters is performed because it is not suitable for the next preprocessing stages. In Ge’ez writing system, characters with the same sound have different symbols. These different symbols must be considered as similar even if they have effect on meaning. As a result, in this study, some symbols of the same sound were converted to one common form. For example, if the character is one of ዐ,ኣ,ዓ(with the sound a) then it will be changed to their equivalent respective orders of አ, similarly, ሐ,ሓ,ሃ,ኀ and ኃ (all of them with a similar sound, h) then it will be converted to ሀ to make ሀለፈ. By the same token, all orders of ሠ (with the sound s) are changed to their equivalent respective orders of ሰ to make ሰአለ. Generally, we normalize the characters based on the words that we have selected in our study but not for all Ge’ez language words.

2.4.1.2. Tokenization: tokenization is very important to this study. It is the process of breaking sentences into words or tokens. The corpus, which is a set of sentences first tokenized into words. Tokenization is done by identifying with the white spaces, comma (,) and special symbols between the words. All punctuation marks, numbers and special characters are removed from the text before the data is processed. Hence, these punctuation marks don’t have any relevance to identify the meaning of ambigous words using WSD. Therefore except '።' which is used to detect the end of the sentence, all other punctuations are detached from words in tokenization process. Tokenization is used to get context words for disambiguation purpose. For instance, if we have a sentence like ወተርፈ : አዳም : ውስተ : ምድረ : ኤዶም : ወቃየንሰ : ሀለፈ : ወሀደረ : ታህተ : ምስራቀ : ኤዶም።, then tokenized sentence will be ወተርፈ, አዳም, ውስተ, ምድረ, ኤዶም, ወቃየንሰ, ሀለፈ, ወሀደረ, ታህተ, ምስራቀ, ኤዶም, ።

2.4.1.3. Stop word removal: after tokenization, we have removed Ge’ez language stop words, as it has no effect on meaning of the words. In this study, stop word removal is used to remove stop words from the corpus because the absence or presence of these words has no contribution to identify appropriate sense. Not all tokenized words are necessary in this work. For this study, we collected stop words which are conjunctions, prepositions and articles of the language because of absence of standard stop words. For instance, words such as (‘ባህቱ’, ‘እንተ’, ‘ከማሁ’, ‘ኩሉ’, ‘እምዛ’,’እስመ’, ‘ድህረ’, ‘እምነ’). Since stop words do not have significant discriminating powers in the meaning of ambiguous words; we filtered stop-words list to ensure that only content bearing words are included. Nevertheless, stop words like ‘ሀበ’ and ‘መንገለ’ are not removed from the corpus because they have a significant role on the word ‘ሀለፈ’.

2.4.1.4. Stemming: Stemming is the process of reducing morphological variants of words into base or root form. In morphologically rich languages like Ge’ez, a stemmer will lead to significant improvements in WSD systems. In Ge’ez language, there are different terms that are generated from the same root word due to their grammatical use. To create different derivational and inflectional word forms, Ge’ez language makes use of prefixes, suffixes, and infixes. Therefore, those extra words or characters that change the root word to different forms are stemmed from the corpus using the stemming algorithm developed by ourselves which is suitable for the language. This algorithm removes both prefixes and suffixes only since we developed affix removal of the stemmer. Therefore, to get the common form of the ambiguous words we tried to normalize infixes of the root word manually. For Example, an ambiguous word ‘ገብረ’ may become like ‘ገብሩ’, ‘ይገብር’, ‘ይግበር’, etc. after removing both prefixes and suffixes. To make it suitable for machine learning algorithms, we inspected manually all those words into one word which is‘ገብረ’. The same thing was applied for other ambiguous words that are used in this study but not for the context words. The reason behind not applying normalization after stemming for the context words is due to the long time it takes to normalize all the context words that are used in this study.

2.4.2. Transliteration

After the above preprocessing tasks were done for Ge’ez documents that we have collected; transliteration were performed for Ge’ez language documents. It is the representation of the characters of one language by corresponding characters of another language. In this study, the transliteration was accomplished from Ge’ez characters into Latin characters to make documents compatible with the machine learning tool called Weka (Getahun Wassie and Million Meshesha, 2014). Since we selected a machine learning tool (called Weka) for conducting our experiment, we applied transliteration because WEKA platform uses Attribute Relation File Format (ARFF) or Comma Separated Value (CSV). These file formats can be applied after transliteration have been performed in order to make it compatible with the WEKA tool. The transliteration of the Ge’ez corpus was conducted by using System for Ethiopic representation in ASCII (SERA).

2.5. Preparing Datasets

In this study, we used corpus based approach. We prepared a combination of labeled datasets for supervised learning, unlabeled datasets for unsupervised learning and semi-labeled datasets semi-supervised for training. Because corpus-based approach uses both labeled and unlabeled datasets for training and testing.

More number of sentences need not be annotated manually in semi supervised machine learning approach. Instead of this, we select the representative seed examples for each sense of ambiguous words. So to select representative seed examples, labeled and unlabeled data size distribution for training set is typically 85-98% unlabeled datasets; and the rest are for labeled datasets (Mahmoodvand and Hourali, 2017). According to this, we prepared 12% of labeled datasets and 88% of unlabeled datasets for each of the six chosen ambiguous words from the total datasets before clustering. That means a word ‘ገብረ’ has 160 instances, so from this 12% are labeled and 88% are unlableled which becomes 20 instances are labeled and 140 instances are unlabled. When we label seed examples automatically, we applied the following techniques.

2.5.1. Seed Selection Techniques

In this section, the techniques that were applied in this study will be presented. This research was conducted by using corpus based approach. Both labeled and unlabeled documents were used in the semi-supervised approach. The seed selection technique employs the method proposed in (Getahun Wassie and Million Meshesha, 2014). The techniques that were used in this study consist of four steps.

Step 1. Selecting representative seed examples for each class or sense of the ambiguous words: Selecting representative seed examples for each class is effective and those selected seed words are used to label unlabeled documents. Selecting seed words to select representative seed examples for semi-supervised approach is challenging task (Getahun Wassie and Million Meshesha, 2014). Selecting improper seed examples results in poor performance. Improper seed examples can be selected when we tag (label) our datasets randomly by humans. When humans select seed examples randomly, they may select improper seed words which means the selected seed words cannot differentiate the senses (meanings) of the ambiguous word.

To minimize manual selection of limited number of seed examples from the total datasets, we used tree algorithms because tree algorithms represent the related concept of the target word starting from the node. Tree algorithms use information gain as lexical knowledge and information gain can minimize subjectivity problem in manual selection of seed examples (Solomon Mekonnen, 2010). By considering this concept, we used ADTree algorithm to select seed words. We classified our datasets by using ADTree algorithm for each ambiguous words and then tree visualization is performed. After tree visualization, we took seed words scoring high information gain in the tree structure. Those seed words were used for discrimination purpose of the remaining sense of the ambiguous word.

Step 2. Clustering both labeled and unlabeled seed examples using “classes to clusters evaluation” mode: here, we have not used the resulting clusters from the ADTree algorithm for classification. We only use them to identify the cluster of missing instances based on labeled seed examples. After clustering have been done using EM algorithm which has best performance in clustering algorithms in this study, we can see the effect of semi-supervised learning method in our work. The clustering result shows that the class labels of some of the seed examples were misclassified. However, automatic clustering suggests that such label changes were not required because those seeds were labeled with their sense class as the promise they are chosen by experts intentionally.

Step 3. Feature Extraction and Selection:

Feature Selection: The success of machine learning requires instances to be represented using an effective set of features that are correlated with the categories of word senses. For this study, feature selection was performed by preparing a eight-eight window size. Because in our datasets the highest window size is eight. Instances with missing values were also removed from the feature sets. Therefore, feature selection is a data reduction mechanism.

Feature Extraction: feature vector which represents words for each instance of a target word, that means files of comma-separated values, a line in WEKA with an extension of. arff or .csv. These vectors represent a text window surrounding ambiguous word of a eight-eight words in our case.

Step 4. Design of the classifier: Before classifying our datasets using the selected classification algorithms, we labeled our datasets manually depending on the selected seed words. The manually labeled seed examples being are used as cluster labels during clustering of both labeled and unlabeled documents (datasets). Knowing cluster label of each instance becomes important for differentiating the class of missed instances by taking each cluster as a distinctive class. This helps us to label unlabeled instances with their classes.

Table 1: Example of WSD Dataset for Semi-Supervised Learning

LContext3	LContext2	LContext1	Target word	RContext1	RContext2	RContext3	Class
?	?	emeze	Halefe	kaeba	bahere	horu	pass
?	tanesio	emeheya	Halefe	halafa	behera	tirose	pass
halifo	kaeba	tirose	Halefe	wosidona	galila	maekala	pass
?	tanesio	emeheya	Halefe	haba	behera	yehuda	?
reeyo	soba	maseya	Halefe	haba	bitaneya	aseretu	?
ahadu	aseretu	keleetu	Halefe	haba	liqana	kahenate	died
maseyo	halafa	aseretu	Halefe	aseretu	keleetu	aredaihu	died
?	?	sanita	Halefe	hagara	nayene	horu	?
?	?	emeze	Halefe	soba	baseha	gize	?
ahadu	beesi	kebure	Halefe	behera	rehuqa	yenesae	?

? Represents missing value

Table 1 indicates that we prepared our datasets with 18 attributes and two of them are target word and class. The rest 16 attributes are context words that are used to determine the meaning of the ambiguous word which means eight words to the left and eight words to the right of the target word. Attributes that exceeds this size are removed. When we use eight words to the left and eight words to the right, there might be missing values. Those missing values are replaced by question marks (?), because question mark is compatible with Weka. Reducing dimensionality of datasets can improve the performance of WSD; because instances having redundancy and missing values problems will be reduced (Solomon Mekonnen, 2010).

2.6. Evaluation Techniques

To evaluate performance of clustering and classification algorithms, four different modes are available in WEKA. Those are using training sets, supplied test set, percentage split, classes to clusters evaluation mode and cross validation for both clustering and classification algorithms. However, when we train and test our datasets on all the above evaluation modes, classes to clusters evaluation mode and cross validation are the most effective evaluation modes. Therefore, for our study we used 'classes to clusters evaluation' mode for clustering algorithm and cross validation for classification algorithm. Therefore, for this study we used 'classes to clusters evaluation' mode for clustering algorithm and cross validation for classification algorithm.

When we use 'classes to cluster evaluation' mode, WEKA shows the clustering result as error rate using 'classes to clusters evaluation' mode. Therefore, accuracy of clustering algorithms was obtained after subtracting the error rate from hundred. This accuracy is used to measure how well it has been able to generalize the clustering results. For this study 10-fold cross-validation evaluation technique is used in our experiment. In this technique, first the total data set is divided into 10 mutually disjoint folds approximately of equal size using stratified sampling mechanism. In stratified sampling, the folds are stratified so that the class distribution of the tuples in each fold is approximately the same as that in the initial data.

We have a total of 2119 manually tagged sense examples which is divided into 10 approximately of equal sizes. As a result of this, each fold of a data set contains 212 sense examples with balanced distribution number of senses per fold. After identifing and separating the training set and testing set from the total datasets, we remove manually tagged sense examples from test set. During this process 90% of the data is used for training to develop the system whereas the remaining 10% is used for testing the system. The process was repeated ten times. After each training phase, the system was tested on average of 212 Ge'ez sentence. Each of the corresponding training set contains an average of 1907 sentences. The performance of classification algorithms is usually measured by parameters such as accuracy, recall, precision and F-measure. These performance parameters are the functions of the numbers of correctly and incorrectly classified instances which are obtained on the confusion matrix of WEKA output.

3. Results and Discussions

This section presents the performance evaluation of the implemented model. To achieve our objectives, the following experiments or scenarios are considered which are applied on our prepared datasets.

- Comparison of corpus based approaches which are supervised, semi-supervised and un-supervised with different modes;
- Investigating the most effective approach and effective algorithm from the selected algorithms that improve the performance of Ge'ez WSD model;
- Experimenting with different context window size for disambiguation of ambiguous words.

3.1. Comparison of Corpus Based Approaches

To compare results of unsupervised, supervised, and semi-supervised machine learning approaches; we used the same datasets of the language, and the classification algorithms for both semi-supervised and supervised approaches. But in study (Solomon Mekonnen, 2010) for unsupervised approach we used the selected clustering algorithms which are Expectation maximization, Simple K-Means, Farthest First, Hierarchical Clusterer for clustering purpose of our datasets. We used clustering algorithms for un-supervised machine learning approach because clustering is unsupervised technique. Therefore, comparison of those three machine learning approaches was conducted on the same datasets. In addition to this, comparison of semi-supervised and supervised approaches was conducted on the same datasets by using the same classification algorithms. The algorithms are SMO, Naïve Bayes and Bagging which are supervised learning methods, and Adaboost and ADtree are used from semi-supervised learning approach.

3.1.1. Unsupervised Learning

Unsupervised learning is an independent process where no supervision is involved during the learning step. Unsupervised corpus based methods do not rely on external knowledge sources such as MRD, concept hierarchies and sense tagged texts. Those approaches are mainly clustering approaches where words and contexts are clustered. During clustering, each cluster corresponds to a sense of a target word. The goal of clustering is to group together elements in a way which maximizes similarity between elements in one cluster and to minimize similarity between elements belonging to different clusters.

3.1.2. Supervised Learning

Supervised is the use of algorithms that reason from externally supplied instances (training set) to form classes to differentiate new data. The goal of supervised learning is to build a model of the distribution of class labels in terms of predictor features. In order to build the model it involves training and testing phases. During the training phase a sense-annotated training corpus is required, from which syntactic and semantic features are extracted to build a classifier using machine learning techniques and in testing phase the classifier tries to find out the appropriate sense for the word based on surrounding words present in the instances.

3.1.3. Semi-Supervised Learning

Semi-supervised techniques involve training information like in supervised but the information given at initial training phase is less. Semi-supervised or minimally supervised methods are gaining popularity because of their ability to get by with only a small amount of annotated reference data while often outperforming totally unsupervised methods on large data sets. There are a host of diverse methods and approaches, which learn important characteristics from auxiliary data and cluster or annotated data using acquired information. For comparison purpose, we can take the maximum average performance or accuracy of the three machine learning approaches by using their best performing algorithms since they record best accuracy for all unsupervised, supervised and semi-supervised methods.The result is shown in Figure 2.

From Figure 2, we can observe that semi-supervised machine learning approach achieves the highest accuracy of WSD prototype models. By using both labeled and unlabeled datasets, the performance of WSD prototype model have improved compared with other approaches. This is because unlabeled datasets are clustered using manually labeled datasets during clustering. From this we can see that semi-supervised machine learning methods are

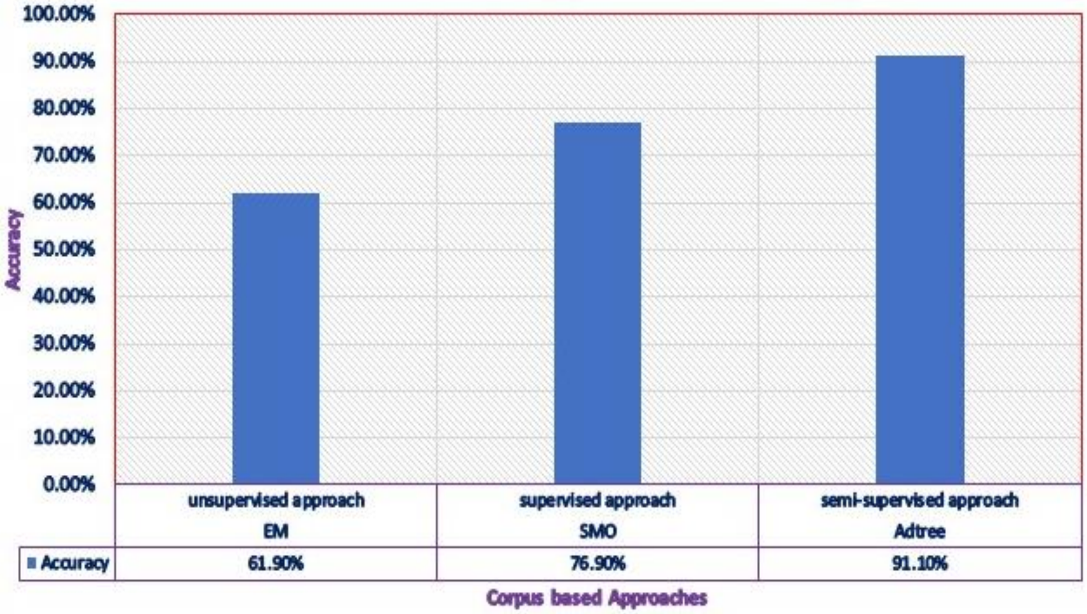


Figure 2: Average performance of the three-machine learning approaches

the most suitable methods for the development of Ge’ez WSD prototype model than supervised and un-supervised machine learning methods using bootstrapping, which means using ADTree, AdaBoostM1 and SMO algorithms. We achieved the best performance of the classifier for Ge’ez ambiguous words using semi-supervised corpus based approach, because seed words which are obtained with high information gain using ADTree algorithm were used for the selection of representative seed examples of this study.

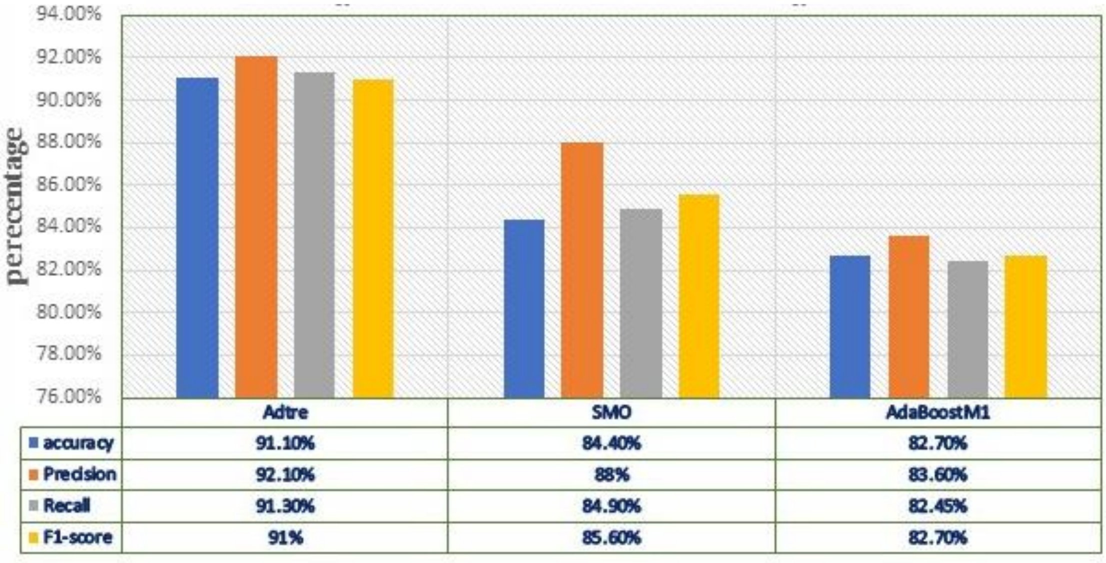
3.2. Comparison of classification algorithm for Ge’ez datasets

For investigating the best performing classification and clustering algorithm for Ge’ez WSD prototype model, we applied three approaches namely unsupervised, supervised and semi-supervised approaches for the selected six ambiguous words of Ge’ez language. We used the result achieved by using semi-supervised methods because the performances achieved by using those machine learning method were the most preferable when we compared with the performances achieved by unsupervised and supervised learning methods. For investigation purpose of those selected semi-supervised algorithms, we used average accuracy, precision, recall and F1-score to access the performance of the three machine learning algorithms. The comparison of those three classifying algorithms was based on the achieved performance to classify ambiguous words of Ge’ez language. Those performance comparison of the selected classification algorithms was done on the same Ge’ez dataset. The result is shown in Figure 3.

From Figure 3, we observe that ADTree algorithm achieves the best performance for our datasets. We achieved an average Precision, Recall, F1-score and Accuracy of 92.1%, 91.3%, 91% and 91.1%, respectively. Its efficiency was also better than AdaBoostM1 and SMO algorithm. AdaBoostM1 and SMO algorithms also performed comparable result to each other for Ge’ez WSD prototype model.

3.3. Determining the optimal context window size of the Language

To find the optimal context window size, different studies have been conducted using different WSD approaches for different languages. WSD researches were conducted for Amharic language by different researchers starting from one-one to ten-ten window sizes to find out the optimal context window size for this language using different approaches (Solomon Mekonnen, 2010). In a research conducted for Amharic language using supervised machine learning method for five ambiguous words (mesasat, meTrat, qereSe, Atena and mesal) and it was advised that window size 3-3 is an effective by using Naïve Bayes algorithm. The authors of (Getahun Wassie and Million Meshesha, 2014) have done a research on Amharic WSD using semi-supervised machine learning method and advised that the optimal window size is 2-2 or 3-3 window size using five classification algorithms for the selected five ambiguous words of the language (ATena, derese, tenesa, ale, bela).



	Adtre	SMO	AdaBoostM1
■ accuracy	91.10%	84.40%	82.70%
■ Precision	92.10%	88%	83.60%
■ Recall	91.30%	84.90%	82.45%
■ F1-score	91%	85.60%	82.70%

Figure 3: Performance of the classification algorithm

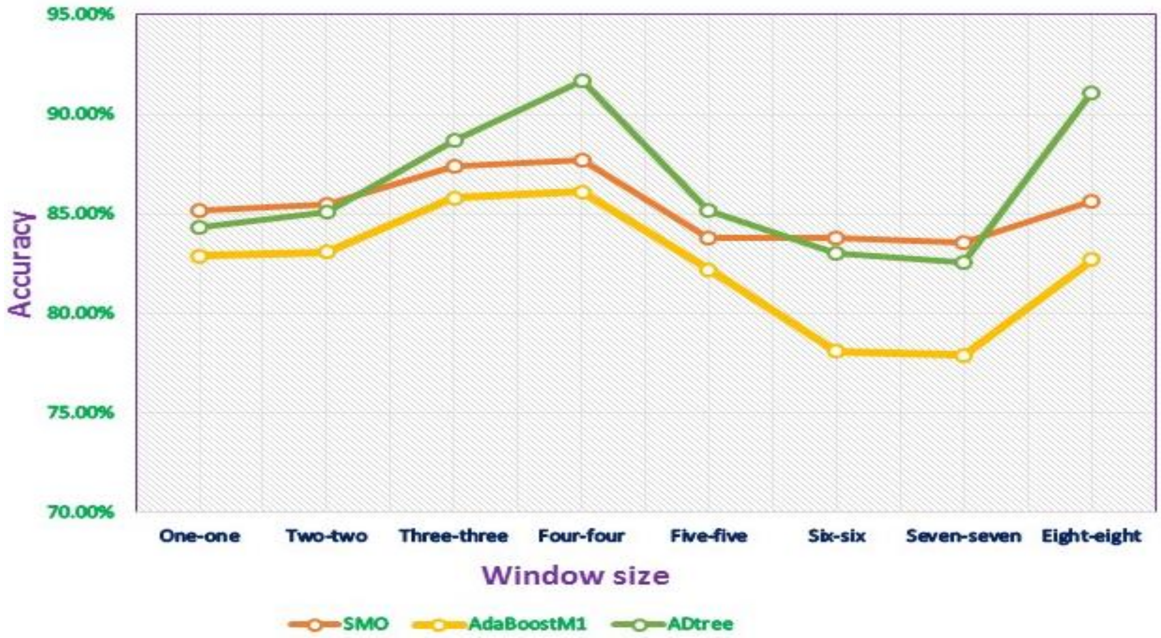


Figure 4: Average Accuracy of each window size using semi-supervised approach algorithms

The optimal window size 3-3 was effective for three of the bootstrapping and SVM algorithms (ADTree, AdaBoostM1, bagging, and SMO), and 2-2 window size was reported to be effective using Naïve Bayes algorithm. Due to those reason we used semi-supervised algorithms to determine the size of context window because Semi-supervised approach scores the highest accuracy than the others approaches. We obtained that window size of 4-4 is the optimal window size in order to differentiate the meaning of the selected Ge'ez ambiguous words using AdaBoostM1 algorithm in our study.

From Figure 4, we concluded that, semi-supervised algorithms perform much better than the other algorithms which means bootstrapping algorithms (ADTree, and AdaBoostM1) SVM algorithm (SMO). We see that ADTree, AdaBoostM1 and SMO achieved high performance on the given datasets. However, our focus is to determine which window size is suitable for Ge'ez language. Then all algorithms that are ADtree, AdaBoostM1 and SMO score high accuracy on window size of 4-4. Therefore, window size 4-4 is the best window size using ADTree algorithm for our Ge'ez datasets. SMO was also the best performer algorithm next to ADTree algorithm for window size of 4-4 using semi-supervised learning method. Lastly, we can conclude that window size of 4-4 becomes best performer using ADTree algorithm for WSD prototype model of our Ge'ez datasets depending on our experiments.

4. Conclusion and Recommendations

4.1. Conclusion

There are so many words with more than one meaning in natural language and the meaning is determined by its context. The automated process of recognizing word senses in context is known as Word Sense Disambiguation (WSD). In this study, three experiments have been conducted using different classification and clustering algorithms.

The first experiment was a comparison of the results obtained using the three machine learning methods which means unsupervised, semi-supervised, and supervised learning methods. In this experiment, a semi-supervised approach has performed better compared to the other machine learning methods. Since the semi-supervised learning method was employed in this work, we used the final fully labeled dataset which is obtained from unlabeled data sets. Those unlabeled datasets were labeled after clustering using the clustering assumption. But for clustering purposes, we used the EM algorithm because the EM algorithm performs better compared to the other selected clustering algorithms. The second experiment was conducted to determine the best performing algorithm for the selected Ge'ez datasets. From this end the best performing algorithm for the selected Ge’ez datasets were found to be the ADTree algorithm compared to both clustering and classification algorithms that were selected in this study. The last experiment was conducted to investigate the optimal window size for determining the senses of each ambiguous word. From experimental results, we obtained that the window size of 4-4 can be considered as optimal window size for Ge’ez WSD systems. In general, we conclude that semi-supervised learning is potential learning method that performs better in our study. There are many potential algorithms to be applied for Ge’ez WSD systems using semi-supervised learning corpus-based approaches and ADtree is the best performing algorithm for this language among those semi-supervised algorithms.

4.2. Recommendations

Word sense disambiguation researches require a variety of linguistic resources like thesaurus, WordNet, Machine-Readable dictionaries and effective Ge’ez language stemmer. There is no standard stop word of the language in which we faced a significant challenge as Ge’ez lacks those resources. Lack of sense annotated data for the language was also another challenge that makes us to limit our study on six ambiguous words of the language. And this makes us to limit our dataset on 2119 sentences or instances only. Therefore, we have the following recommendations which include the development of resources and future research directions for WSD of Ge’ez language:

- This study considers words that have only two senses. In the futures, the researcher will consider words with more than two senses.
- This study has concentrated only on modeling WSD to tackle lexical ambiguity which is at word level. Further researches would be recommended to address other types of ambiguities in Ge’ez language like Character and structural ambiguity (sentsnces level).
- In addition to the corpus-based approaches, there are also knowledge-based and hybrid approaches that were used for WSD of other languages. Therefore, we recommend that these approaches need to be investigated for Ge’ez language as well.

Referencec

Eker, Ö. (2007). Developing Methods For Word Sense Disambiguation. Boğaziçi University.

Getahun Wassie, Ramesh, B. P., Solomon Teferra, & Million Meshesha. (2014). A Word Sense Disambiguation Model for Amharic Words using Semi-Supervised Learning Paradigm. Science, *Technology and Arts Research Journal*, *3*(3): 147-155.

Jurafsky, D. (2000). Speech & language processing. Pearson Education India.

Leykun Berhanu(2005). Contemporary Challenges in the Ministry of the Ethiopian Orthodox Church. PhD thesis, Howard University.

Mahmoodvand, M., & Hourali, M. (2017). Semi-supervised approach for Persian word sense disambiguation. In 2017 7th International Conference on Computer and Knowledge Engineering (ICCKE), 104-110.

Mersa Mebrhatu (2018). Unsupervised Machine Learning Approach for Tigrigna Word Sense Disambiguation. Computer Engineering and Intelligent Systems, *9*(6): 10–16.

Naseer, A., & Hussain, S. (2009). Supervised word sense disambiguation for Urdu using Bayesian classification. Center for Research in Urdu Language Processing, Lahore, Pakistan.

Pal, A. R., Kundu, A., Singh, A., Shekhar, R., & Sinha, K. (2013). A Hybrid Approach To W ord Sense Disambiguation Combining Supervised And Unsupervised Learning. 4(4): 89–101.

Seid Yesuf & Yaregal Assabie (2017). Amharic Word Sense Disambiguation Using Wordnet. In The 5th International Conference on the Advancement of Science and Technology.

Solomon Mekonen. (2010). Word Sense Disambiguation for Amharic Text: A Machine Learning Approach. Unpublished Master’s Thesis, 1-94.

Workneh Tesema, Tesfaye Debela & Kibebew Teferi.(2016). Towards the sense disambiguation of Afan Oromo words using hybrid approach (unsupervised machine learning and rule based). *Ethiopian Journal of Education and Sciences*, 12(1): 61-77.