Comparative Analysis between Naïve Bayes Algorithm and Decision Tree to Solve WSD Using Empirical Approach

Boshra F. Zopon Al-Bayaty and Shashank Joshi

Abstract—There are many robust approaches algorithms for Word Sense Disambiguation using machine learning, and it is very difficult to make comparisons between them if we don’t implementation empirically. In this word, analysis and developed JAVA Code and compare between two of the most successfully approaches for supervised machine learning, namely, Naïve Bayes and Decision tree using WordNet and Senseval for Word Sense Disambiguation of words in context.

While comparing these two approaches, paper deals with supervised learning method proving effective results. These algorithms refer common data set, training file and testing file to calculate accuracy to predict exact meaning of sense.

Index Terms —Supervised, Naïve bayes, decision tree, WSD, WordNet.

I. INTRODUCTION

In last year’s, there are many researchers widely search empirically in NLP field, and WSD problem. The task remove the ambiguity from the words and select proper sense called WSD, and this task require examine the word in context and determine which sense can be use. There are many words have multiple meaning according to the context of speech. For example the word (Recompense) has different meaning in context, as in screenshot below in Fig. 1 [1]:

![Fig. 1. The screenshot from WordNet shows the multiple of recompense word.](image)

Fig. 1. The screenshot from WordNet shows the multiple of recompense word.

The experiments in this field proved there are many methods can used it and adopt it in this domain, by analysis and test each of them empirically, to prove the objectives of research successfully or not.

Our goal is remove the ambiguity from the word by select the correct sense that annotated from WordNet. Word sense disambiguation is the task that examines the word in context and selects the proper meaning among many senses or meaning related with the word. WSD task so important for many purposes in natural language processing, like, machine learning, information retrieval, and so on of natural language processing purposes [2].

This study is one of the experiments of our PhD research work currently, to complete master- slave technique [3]. We focused in this paper on two of supervised methods, one of them decision tree which based on classifications rules, and the second one Naïve bayes, one of probabilistic learning methods. We presented analysis and comparison between these two supervised learning approaches stared from selection data set, training, testing data, till calculate the results of them [4].

However, now days the word sense disambigation still open problem in natural language processing domain, and there is a scope to enhancement the accuracy of selection proper sense [5].

II. MOTIVATION AND APPLICATION

Where a input is accepted and perception of user influence the result to be displayed especially search engine which displays result after accepting input from user. Every domain which works on same concept where input is accepted to deliver output according to the result. Every NLP application where result could be affected by correct or incorrect interpretation[5].

There are many applications for word sense disambiguation such as:

**Information Retrieval** [6]: Data is retrieved by using search engine or likewise interface if we do not mention sense. Especially there is no way that system can analyze it meaning without help of WSD technique. There are many examples which would lead to incorrect meaning of given word, like [bank, finance related bank], [bank, Edge of river].

**Machine translation**: Convey information to machine correctly, so that further conversion using intermediate code could be carried out; for example, flexographic conversion.

**Phonetics**: Disambiguation could occur not only in NLP but also at domain where there is conversion from speech to text. For example, [eat – to consume food or so, it \(\rightarrow\) pronoun, pronoun station could lead to different text and there by
different meaning.

III. RELATED WORK

Word sense disambiguation one of the open problem in NLP, plenty of work is carried out to solve this problem, but there is lot of scope to contribute in this field to identify sense of given word correctly. Generally disambiguation is resolved by using many approaches, the main approaches include [7], [8]:

Supervised Approaches: Where system is trained to correctly identify meaning of particular word.

Unsupervised Approaches: Based on the group or collection of required data result is fetched.

There are many robust algorithms like Naïve bayes, SVM, decision tree, decision list, KNN, and so on which could be used to address word sense disambiguation.

IV. SUPERVISED MACHINE LEARNING APPROACHES

Machine learning approaches can be used to discover the relationships in the internal structure of the data and production outputs are correct. These approaches composed Naïve bayes, decision list, decision tree, support vector machines and some of supervised machine learning methods.

In this work we tried to do a comparison between the well known supervised learning approaches, Naïve bayes and decision tree which both have long successful history in this field. Naïve bayes is the most commonly approach used in Word Sense Disambiguation, we have implemented the algorithms using WordNet 2.1, and our study to Naïve Bayes achieved (62.86 %) accuracy to the Senseval-3 [9]. And according to results from implementation decision tree we achieved (45.14%) accuracy [10].

Our goal is to see which one is the most successful in performance through a comparison between the two algorithms and study the factors affecting them and the possibility of improving the performance of each and improve the accuracy, by combining them together in future, showing in Table I.

### TABLE I: NAÏVE BAYES ATTRIBUTE EXAMPLE

<table>
<thead>
<tr>
<th>Attribute</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>3 MP Camera, 5 inch screen, 50 gm weight, black color</td>
<td>10 MP Camera, 4.8 inch screen water proof, purple color</td>
<td>8 MP Camera, 6 inch screen, 100 gm weight, white color</td>
</tr>
</tbody>
</table>

A. Naïve Bayes Approach

This approach is one of the important algorithms used in data mining. It is based on conditional probability. In naïve bayes algorithm information about various objects and their properties is collected during the training phase system is trained to identify new object based on respective attributes of these objects. For example selection of mobile this is added and identify. Its category based on the information available related with attributes. Consider a scenario where three mobiles, X, Y, Z are described [11].

System is trained with this information and when we want identify any new Mobile individual attributes are evaluated and match is found.

For implementing WordNet data source is used this is repository which provides the mapping of word and different sense associated with that word. For performing on experiment we referred data set 10 nouns and 5 verbs which contain following words [12]:

Data set of pos (n) = {Praise, Name, Lord, Worlds, Owner, Recompense, Straight, Path, Anger, Day}. Data set of pos (v) = {Worship, Rely, Guide, Favored, Help}.

To use WordNet repository senseval XML mapping technique is used [13], where the given data set and senses are expressed with XML. And to ensure effective working of decision tree training and testing file is used. Job of file is to provide the context which will be extremely useful exactly know meaning of particular word. For implementing C4.5 algorithm eclipse ID2, is used, while implementing it equations related with entropy are implemented. Below the algorithm we applied:

1. Initialize context c, sense s, and ambiguous word w.
2. As per training context
3. \[ p(s \mid w, c) = \frac{p(w \mid s, c) p(s \mid c)}{p(w \mid c)} \]

Calculate Maximized \( p(s \mid w, c) \)
4. Select one with highest value
5. Map sense according to the highest accuracy.

Box 1. Naïve bayes algorithm implemented on our data set.

1) Naïve bayes network

In this section, Naïve Bayesian classifier has been implemented for instance word “Path” from our data set with the four senses (s), the calculations involved as mentioned in Fig. 2:

\[
P (s_1) = \frac{3}{14} = 0.214 \\
P (s_2) = \frac{2}{14} = 0.142 \\
P (s_3) = \frac{5}{14} = 0.357 \\
P (s_4) = \frac{4}{14} = 0.285
\]

Consider four different words selected from bag of words –
(1.(f1) Travel, 2.(f2) Life, 3.(f3)Course, 4.(f4) Which. For sense s3-“pattern”:-

\[ F_1 = \frac{2}{14} = 0.142, \]
\[ F_2 = \frac{1}{14} = 0.0714, \]
\[ F_3 = \frac{2}{14} = 0.142, \]
\[ F_4 = \frac{3}{14} = 0.214. \]

For designing baysian Network:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Decision</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rain</td>
<td>Yes</td>
<td>Less temp</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>More temp</td>
</tr>
<tr>
<td>Temperature</td>
<td>Yes</td>
<td>Less humidity</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>More humidity</td>
</tr>
<tr>
<td>Humidity</td>
<td>Yes</td>
<td>Less temp</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>More temp</td>
</tr>
</tbody>
</table>

Baysian networks gives details about contribution of individual feature and one of the sense from number of senses available of given word. Word sense feature (combines the baysian network). It helps to check individual assessment of every feature (F1, F2, F3, F4), seeing in Fig. 3.

\[ S=\text{pattern} \]

\[ F_1 = \text{Travel} \]
\[ F_2 = \text{Life} \]
\[ F_3 = \text{Course} \]
\[ F_4 = \text{which} \]

Fig. 3. The naive bayesian network.

B. Decision Tree

1. Read data set and calculation POS (e.g. recompense.)
2. Prepare context containing various senses of word (e.g. Recompense - reward)
3. Calculate frequency at context (i.e. - P- and +P+)
   - P- Negative
   - P+ Positive
4. Calculate information gain for calculating entropy (S) = - P+logP+ - P-logP-
5. Gain (S,A) = Entropy(S) - \( \sum_{v \in D} \left| S_v \right| \) Entropy (Sv)
6. Select highest (Entropy, Attribute ratio)
7. E.g. (S,A) for recompense = 0.593
   - For = reward

Box 2. C4.5 algorithm implemented on our data set.

Decision tree is a predicative model, which helps to take decision on the statistics’ available (past information). In a decision tree branches provides attribute or related condition on which decision is made in the form of nodes (Yes or No) [14]. If clear decision is not made by branches then information gain is checked whichever nod has high information gain that node is declared as correct or final decision. In C4.5 algorithm every time information gain is calculated for entropy which is useful in making decisions.

Consider a simple example of whether for casting in which decisions are made or predicated to remove uncertainty. If clouds are dense in sky there will be rain. If there is rain then temperature will get decreased and humidity will get increased [15].

In this decisions can be made bored on the available information to decide the destination on the basis of highest value of information gain. Box 2. Shows the algorithm we applied in our study.

V. WORD SENSE DISAMBIGUATION EXPERIMENTS WITH NAÏVE BAYES AND DECISION TREE

A. Dataset

We started with dataset provided by the http://www.e-quran.com/language/english, the dataset composed 15 English words, 10 nouns and 5 verbs, such as path, help which are have ambiguity. Since one of particular steps are goal is to train the dataset and to disambiguate the words by selecting the proper meaning in context [16], we used WordNet, which is available at http://wordnet.princeton.edu, to provide the sense of words information’s. And to make sure, test, evaluation the both approaches and properly assigned to word, we used practically senseval-3 in empirical our work, seeing in Table II.

TABLE II: DECISION TREE ATTRIBUTE EXAMPLE

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Decision</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rain</td>
<td>Yes</td>
<td>Less temp</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>More temp</td>
</tr>
<tr>
<td>Temperature</td>
<td>Yes</td>
<td>Less humidity</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>More humidity</td>
</tr>
<tr>
<td>Humidity</td>
<td>Yes</td>
<td>Less temp</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>More temp</td>
</tr>
</tbody>
</table>

B. Analysis

Results acquired by naïve bayes approach and decision tree approaches are compared for some cases, Naïve bayes approach gives better result and other decision tree is more efficient. If size of tree is less then decision tree gives better result. Overall accuracy of Naïve bayes accuracy of decision tree.

C. Modeling

Application is made up of number of modules some of imp.mod areas below [17]:
1) Dictionary ➔ Data source.
2) Training ➔ context providing base for context.
3) Testing ➔ verification of data and its meaning.
4) Sense Map ➔ Mapping between word and sense.
Apart from this there are, many packages, classes calculate accuracy of sense.

D. Design

To address word sense disambiguation semi-structured data is used to enhance the performance. Algorithm along with given context will train system to judge correct sense, which is further verified by the testing file to ensure correct meaning of sense, seeing in Table III.

E. Training

Data set of 10 nouns and 5 verbs is used. To make understanding of senses, system is trained by referring senseval-3 structure to map word with sense by using surrounding context. This entire structure uses XML format to represent and process data using semi structured approach.

<table>
<thead>
<tr>
<th>Word</th>
<th>POS</th>
<th># Sense</th>
<th>Naïve Bayes Score</th>
<th>Naïve Bayes Accuracy</th>
<th>Decision Tree Score</th>
<th>Decision Tree Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Praise</td>
<td>n</td>
<td>2</td>
<td>0.408</td>
<td>0.592</td>
<td>405</td>
<td>593</td>
</tr>
<tr>
<td>Name</td>
<td>n</td>
<td>6</td>
<td>0.189</td>
<td>1.0</td>
<td>184</td>
<td>1000</td>
</tr>
<tr>
<td>Worship</td>
<td>v</td>
<td>3</td>
<td>0.172</td>
<td>0.414</td>
<td>308</td>
<td>425</td>
</tr>
<tr>
<td>Worlds</td>
<td>n</td>
<td>8</td>
<td>0.137</td>
<td>1.0</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>Lord</td>
<td>n</td>
<td>3</td>
<td>0.341</td>
<td>0.681</td>
<td>187</td>
<td>426</td>
</tr>
<tr>
<td>Owner</td>
<td>n</td>
<td>2</td>
<td>0.406</td>
<td>0.594</td>
<td>405</td>
<td>595</td>
</tr>
<tr>
<td>Recompense</td>
<td>n</td>
<td>2</td>
<td>0.48</td>
<td>0.594</td>
<td>405</td>
<td>595</td>
</tr>
<tr>
<td>Trust</td>
<td>v</td>
<td>6</td>
<td>0.167</td>
<td>0.167</td>
<td>167</td>
<td>167</td>
</tr>
<tr>
<td>Guide</td>
<td>v</td>
<td>5</td>
<td>0.352</td>
<td>0.648</td>
<td>199</td>
<td>247</td>
</tr>
<tr>
<td>Straight</td>
<td>n</td>
<td>3</td>
<td>0.496</td>
<td>0.504</td>
<td>462</td>
<td>462</td>
</tr>
<tr>
<td>Path</td>
<td>n</td>
<td>4</td>
<td>0.415</td>
<td>0.585</td>
<td>316</td>
<td>316</td>
</tr>
<tr>
<td>anger</td>
<td>n</td>
<td>3</td>
<td>0.412</td>
<td>0.588</td>
<td>462</td>
<td>462</td>
</tr>
<tr>
<td>Day</td>
<td>n</td>
<td>10</td>
<td>0.109</td>
<td>1.0</td>
<td>109</td>
<td>109</td>
</tr>
<tr>
<td>Favored</td>
<td>v</td>
<td>4</td>
<td>0.587</td>
<td>0.648</td>
<td>250</td>
<td>250</td>
</tr>
<tr>
<td>Help</td>
<td>v</td>
<td>8</td>
<td>0.352</td>
<td>0.414</td>
<td>125</td>
<td>125</td>
</tr>
</tbody>
</table>

F. Testing

Given data is tested with XML file which contain context without direct mapping with sense. This approach results in accurate prediction of sense for given word [18], seeing in Fig. 4.

G. The Execution Steps

We brief our execution steps as blow:

Data Source: Decide suitable data to be checked for WDS. Select sample words to check behavior of algorithm. In the experiment 10 noun and 5 verbs are used.

Dictionary: Refer format at senseval. Org. Prepare XML format of content helping to resolve sense of data with respect to some unique ID [19].

Algorithm: Write a code to check accuracy of the word to predict exact sense by referring the given context for a data set selected as mentioned above.

Execution this algorithms in eclipse kepler to get score made for given sense, select the sense with high accuracy as a final result.

H. The System Answer

Results of word sense disambiguation are stored in a file called as system Answer.txt. This file displays the score for respective senses of word in given dataset [20].

The said score is calculated on the scale of 1000. Sense having highest score of accuracy is considered as correct sense identification. After performing the experiment overall accuracy of Naïve Bayes algorithm is (62.86 %), and Decision tree is (45.14 %). This accuracy is calculated on a data set of 10 nouns and 5 verbs on the basis of context to resolve the meaning of a word.
Result shows for some words, Naïve Bayes algorithm provides better results for example [Name, Worlds, and Day], for other case decision tree provide better accuracy values for example [Name and worlds only].

The screenshot below shows the System Answer. Txt file for decision tree implemented.

VI. THE FINAL RESULT

Mention accuracy with high values using decision tree and Naïve bayes approach. As a sum of overall accuracy Naïve Bayes approach gives more accurate results, as shown in Table IV and Fig. 5:

![Image](image_url)

**TABLE IV: THE FINAL RESULTS OF NAÏVE BAYES AND DECISION TREE CLASSIFIERS**

<table>
<thead>
<tr>
<th>Approaches</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naïve Bayes</td>
<td>62.86</td>
</tr>
<tr>
<td>Decision Tree</td>
<td>45.14</td>
</tr>
</tbody>
</table>

VII. CONCLUSION

There cannot be a 100% accurate method. It depends upon data set context and algorithm we used to implement Word Sense Disambiguation. Accuracy, likely to vary according to these parameters. Still Naïve Bayes approach gives more accurate result in same time as per experiment. Table IV Below shows the final results of accuracy for both approaches.

ACKNOWLEDGMENT

The first author thanks the research guide Dr. Shashank Joshi (Professor at Bharati Vidyapeeth University, College of Engineering) for submitted his advices within preparing this paper.

REFERENCES

[16] M. Joshi, T. Pedersen and R. Machin, “A combative study of support vector machines applied to he supervised word sense disambiguation problem in the medical domain,” Department of Computer Science, University of Minnesota, Duluth, MN 55812, USA.

Boshra F. Zopon Al-Bayaty received her B.E degree in computer science from Al-Mustansiriya University, College of Education in 2002. She received her M.S.C degree in computer science from Iraqi Commission for Computers and Informatics, Informatics Institute for Postgraduate Studies. She is doing her the Ph.D. in computer science at Bharati Vidyapeeth Deemed University, Pune.

She is currently working in the Ministry of Higher Education & Scientific Research, Al-Mustansiriya University in Iraq. Baghdad. Her research interest is focused on software engineering.

Shashank Joshi received his B.E. degree in electronics and telecommunication from Govt. College of Engineering, Pune in 1988, the M.E. and Ph. D. degrees in computer engineering from Bharati Vidyapeeth Deemed University Pune. He is currently working as the professor in Computer Engineering Department at Bharati Vidyapeeth Deemed University College of Engineering, Pune. His research interests include software engineering. Presently he is engaged in SDLC and secures software development methodologies. He is an innovative teacher devoted to education and learning for the last 23 years.