A Comparative Study of Two Different Bag of Words Data Representations for Urdu Word Sense Disambiguation

Hamad Ahmed[a], Omar Salman[a], Ehsan-ul-Haq[a], Kashif Javed[a], Sana Shams[a], Sarmad Hussain[a]

[a] *Center for Language Engineering, Al-Khawarizmi Institute of Computer Science, University of Engineering & Technology, Lahore 54890, Pakistan*

Abstract

This paper compares the accuracies of two different data representation techniques: 'Bag of Words in a Complete Sentence' and 'Bag of Words in a Limited Size Window' for the Word Sense Disambiguation(WSD) problem in Urdu language. In languages like English, Hindi, Persian etc. higher accuracy has been reported by using Bag of Words in a Limited Size Window as compared to a complete sentence. Urdu is, however, unique from other languages in several linguistic aspects and the same facts cannot be readily generalized for it. We tested the two data representations using Naïve Bayes and Support Vector Machines classifiers on sets of 11 Urdu words. Results showed that Bag of Words in a Complete Sentence completely dominates the Bag of Words in a Limited Size Window representation indicating that Urdu words need more contextual information for sense discrimination.

Keywords: Word Sense Disambiguation, Data Representations, Bag of Words, Supervised Machine Learning, Naive Bayes, Support Vector Machines

1. Introduction

Words are polysemous, their correct meaning can only be inferred from the context in which they occur. For example, the word 'bank' is used as a financial establishment in 'He deposited his money in the bank' and as a side of a river in 'He went to the nearby bank for fishing'. The context of a word is indicative

Email addresses: hammad.rana@hotmail.com (Hamad Ahmed), omar.salman@kics.edu.pk (Omar Salman), ehsan.ulhaq@kics.edu.pk (Ehsan-ul-Haq), kashif.javed@kics.edu.pk (Kashif Javed), sana.shams@kics.edu.pk (Sana Shams), sarmad.hussain@kics.edu.pk (Sarmad Hussain)

Corresponding Author: Hamad Ahmed, *Email Address:* Same as above, *Postal Address:* Same as institute, *Contact No.:* +92-331-4475244

of its true meaning. Such multi-sense words are present in all languages. In English, the 121 most high frequency nouns have an average of 7.8 meanings per word[1]. Urdu language is also rich in these words e.g. the word 'زبان' (zəba:n) can have two meanings, ‘tongue’ or ‘language’. The word 'حصہ' (hsə) can have two meanings, ‘take part in’ or ‘be a part of’. We use these words very frequently in our daily life communication, however, WSD is a process that comes naturally to humans and we seldom notice how our mind perceives the correct sense of a word from its context[2]. Training a machine to do the same is, however, a challenging task and the focus of this research. Word sense disambiguation is an integral component of machine translation[3], question answer systems[4], information retrieval[5] and language processing[6].

Broadly, three main approaches are used for performing WSD. The first one is supervised learning which uses manually sense tagged Corpora to train classifiers for classifying new instances[7]. The second one is unsupervised learning which does not use any sense tagged evidence, instead it clusters sentences into groups based on the similarities in their feature set[8]. The third one is knowledge based approach which uses a dictionary, thesaurus or any other large knowledge source to find the relations of a word in a sentence with its meaning and gloss in the respective knowledge resource[9].This paper focuses only on supervised learning for WSD.

Supervised machine learning solves WSD by observing the context of the target word. The most important parameter is the size of the context, i.e. N words around the target word, that can effectively capture and indicate its true meaning. These context words form the feature set which is input to the machine learning algorithms. There exist several different approaches for formulating this feature set. These approaches can be thought of as different forms of data representations of the context words, and they are categorized into two major categories: Collocation and Bag of Words[6]. In collocation, we capture information specific to the position of the context words with respect to the target word e.g. information about the word just after the target word, the word exactly four spaces to the right of the target word. The information can include properties such as word roots, parts of speech etc. Thus collocation works on lexically rich information that is very specific to the position of the words. Bag of Words, on the other hand consists of an unordered set of words occurring on either side of the target word in a window of a certain size. WSD in time critical applications like search engines and online dictionaries would suffer from significant delays with collocations since a parser has to annotate the words of the sentence with the appropriate lexical features according to the algorithm before any classification can be done. Bag of words on the other hand, is a simple representation, a container for all the context words which is both space and time efficient. In addition to WSD, bag of words data representation is also used in other Natural Language Processing (NLP) tasks, and forms the basis of modern search engines [6].

Much progress has been made regarding WSD in languages like English, Japanese and Chinese. Generally higher accuracy in WSD is achieved if a large sense-annotated corpus is available for training the classifiers[10]. The SEMCOR (Semantic Concordance) corpus [11] for English, the Japanese SEMCOR [12] for Japanese and a large scale sense annotated Chinese corpus [13] have been prepared which are used for WSD in these languages. Also the algorithms and techniques for achieving higher classification accuracy have been discovered and explored. Urdu language on the other hand, suffers from lack of such resources and preliminary work in this area. A limited sense tagged corpus called the CLE (Center for Language Engineering[16]) Urdu digest corpus [14] has recently been developed which is the first one of its kind. Bayesian classification of word senses has been explored in [15] which is the only work done uptil now for Urdu WSD, making it a relatively newer and challenging field of investigation.

In this paper we explore the bag of words data representation for WSD in Urdu and particularly focus on whether the complete sentence or a limited size window contributes to a higher classification accuracy. We take 11 Urdu words from the sense tagged CLE Urdu digest corpus[17] and use two classifiers, naive bayes and support vector machines on both data representations and compare their accuracies.

The rest of the paper is arranged as follows: Section 2 describes previous studies on WSD and data representations for supervised machine learning for various languages. Section 3 presents the motivation behind this study. Section 4 gives the detailed procedure of the experiments while section 5 presents the results. In Section 6 we discuss the results. This work is concluded in Section 7.

2. Related work

A significant amount of work has been done on supervised machine learning for word sense disambiguation in English. Mihalcea [18] used a window of size 3 to form collocational feature sets for disambiguating 30 polysemous words. He generated the sense-tagged corpus from Wikipedia using hyperlinks of the articles for sense-annotations. He reported 84.9% accuracy using Naïve Bayes classifier. Ng and Lee in [19] explored several different data representations for supervised machine learning including bag of words, POS tagged words, verb-object syntactic relation and collocations on window size 3. They developed a software called LEXAS and achieved a mean accuracy of 87.4%. The same authors in [20] used several different classifiers including Naïve Bayes, SVM, and Decision Trees for testing against each data representation. They reported that different representations give different accuracies with different classifiers. Collocations contributes most to SVM (61.8% accuracy) whereas POS tagged window contribute most to Naïve Bayes. Pederson [21] used bag of words of 9 different sizes i.e. 1, 2, 3, 4, 5, 10, 25, and 50 words on both right and left side of the target word and trained separate Naïve Bayes classifiers for each window size. He then ensembled the 9 classifiers into a single classifier and obtained 89% classification accuracy. Wang et al. [22] used the bag of words model for context representation with a window size of 5. However they ensured that 5 words on either sides were captured by taking words from neighboring sentences if the sentence containing the target word was small. Liu et al. [23] applied supervised learning for disambguating words in English as well as medical terminologies. They used six representations i.e. various combinations of collocations, bag of words, oriented bag of words and five window sizes (2, 4, 6, 8, and 10). They reported the same findings as [20] that different representations contribute more to different classifiers. Hence the main focus has been on using a fixed window size rather than a complete sentence in English because of more accurate results. Considerable amount of work regarding WSD has also been done in other languages. Singh and Siddiqui [24] worked on word sense disambiguation in Hindi and studied the role of semantic relations. They created a context vector by using the bag of words model for limited windows of sizes 5-25. They reported a mean accuracy of 54.5%.

In [25] the authors attempted word sense disambiguation on 20 polysemous Chinese words with 2-8 senses using Chinese Wordnet. They used Bag of Words complete sentence model and improved the classification accuracy from previously best reported 33% to 74%.

In Japanese, [26] demonstrates WSD using a window size of 50 with collocations as well as part of speech tagging. They obtained an overall accuracy of 71.7%. Hamidi et al [27] used bag of words complete sentence model for Persian word sense disambiguation. They used two classifiers Naïve Bayes and k-NN and established the superiority of k-NN classifiers.

As far as Urdu is concerned, the only notable work found in the literature is [15] in which the authors used Naive Bayes classifier for disambiguating 4 Urdu words using limited size window representation. Thus a very limited amount of work has been done in Urdu regarding WSD.

3. Motivation

Urdu has mainly originated from Arabic and Persian with minor influences from Turkish and possesses a character set completely different from English[28]. Apart from a unique character set, several linguistic aspects also differentiate Urdu from other languages. The usual sentence structure for English is subject-verb-object e.g. ‘Ali ate oranges’, whereas Urdu’s sentence structure is subject-object-verb e.g. 'علی نے مالٹے کھائے'(øəl n: ma:l:t: k^{h}a::). This especially impacts the performance of WSD e.g. the English sentence ‘Ali ate oranges after careful examination’ would be written in Urdu as ' علی نے بہت غوروفکر کے بعد مالٹے کھائے`(øəl n: bhət o:ro:fkər k baød ma:l: k^{h}a::) where we see that Ali and oranges occur far apart from each other in Urdu than in English because of the difference in sentence structure. Likewise in English, the prepositions appear before the noun e.g. ‘In the room’ whereas in Urdu, they appear after the noun and can be termed as postpositions e.g. 'کمرے میں' (kəmr: mN̲). Also, Urdu nouns have either a masculine or feminine associated with them and the verbs take on a form with respect to the gender being addressed. For example ‘He eats food’ and ‘She eats food’ would appear in Urdu as 'وہ کھانا کھاتا ہے' (u: k^{h}a:na: k^{h}a:ta: e:) and 'وہ کھانا کھاتی ہے' (u: k^{h}a:na: k^{h}a:ti: e:) respectively. Since word sense disambiguation relies on the context of a target word and the sentence structure dictates the arrangement of these context words in the sentence, this difference in the sentence structure demands investigation of the different Bag of Words models for WSD. Also, the techniques developed for English word sense disambiguation cannot be used for Urdu Word Sense Disambiguation, creating a need for developing separate WSD tools for Urdu.

3.1. Rationale for the current work

The context words help in discovering the true sense of the target word. Not all context words are important in this regard and only a few play the key role of disambiguating the meaning. The location of these key words with respect to the target word is important since an effective window must be long enough to capture all of them. To develop the Urdu WSD, we used two data representation models: ‘Bag of Words in a Complete Sentence’ and ‘Bag of Words in a Limited Size Window’. In those cases where the length of the sentence and the window size are almost equal, no significant difference in performance should be observed since the feature vectors formed by both the models will be the same. However for longer sentences where the length of the sentence is much greater than the window size, the feature vector will be of different lengths and the complete sentence model can be expected to capture more meaningful context words. However chances are also high that it can capture irrelevant information but this issue can be mitigated by applying feature selection which will be described later. Consider the following example for WSD of 'زبان'(zəba:n) which has two meanings: tongue (lets denote it by +1) and language (denote it bye -1):

1. ہم جو بھی گفتگو کرتے ہیں اپنی زبان سے کرتے ہیں۔

(əm do: bʰi: ft̪u: kərt̪: hi:ŋ əpni: zəba:n s: kərt̪: hi:ŋ)

2. دو افراد گفتگو کے لیے ایک ہی زبان استعمال کرتے ہیں۔

(do: :əfra:d ft̪u: k: lj: :i:k i: zəba:n ka: st̪e:m:l kərt̪: hi:ŋ)

3. اکثر دمہ کے مریضوں کی طویل گفتگو کرتے ہوئے زبان باہر آ جاتی ہے۔

(:ksər dəmøkmərzo:ŋ ki: t̪əi:l ft̪u: kərt̪: : zəba:n ba:r a: ja:t̪i: hi:ŋ)

4. ہمارے اعمال کا دارومدار گفتگو کرتے ہوئے اپنی زبان کے صحیح استعمال پر ہے۔

(əma:r: :øma:l ka: da:ro:məda:r ft̪u: kərt̪: : :pni: zəba:n ka: səhh :st̪ma:l hi:ŋ)

5. اس تحریر سے متعلق ایک ماہر ہی بتا سکتا ہے کہ زبان کا استعمال صحیح ہے۔

(:s t̪e:rr s: mt̪:lq ::k ma:r i: bət̪a: səkt̪a: : k zəba:n ka: :st:ma:l sə hi:ŋ)

Lets consider the bag of words window with size 5, Table 1 shows that the feature vector for this representation has only 8 words. It is clear that the words present in the feature vector are common to almost all sentences of both the classes and it is hard to discriminate the sense based on these overlapping words. A linguist can analyze that the key words which help in disambiguating these sentences are present in the corners of the sentence e.g. in sentence 2, 'افراد '(:əfra:d) plays a key role in identifying the correct meaning but it is not present in the feature vector. Similarly, 'دمہ '(dəmø), 'اعمال'((A:øma:l)) and 'تحریر'(t̪e:rr) in sentences 3,4 and 5 are the key words but not present in the feature vector. We can increase the window size so that these words get included in the feature vector

	Sentence 1	Sentence 2	Sentence 3	Sentence 4	Sentence 5
Meaning	+1	-1	+1	+1	-1
گفتگو(ft̪u:)	1	1	1	1	0
کر(kər)	1	1	1	1	0
اپنی(:pni:)	1	0	0	1	0
استعمال(:st̪ma:l)	0	1	1	0	1
باہر(ba:)	0	0	1	0	0
طویل(t̪əi:l)	0	0	1	0	0
صحیح(sə)	0	0	0	1	1

Table 1: Feature vector for Bag of Words Window Size 5 Model

but the position of the key words will vary from sentence to sentence and we will not be able to generalize a window size suitable for all sentences. Also the feature values of the sentences belonging to the same class are not very similar e.g. sentence 1 is more similar to sentence 2 which belongs to the other class, as compared to sentence 3 which belongs to the same class.

Now considering the complete sentence model, Table 2 shows the feature vector.

	Sentence 1	Sentence 2	Sentence 3	Sentence 4	Sentence 5
Meaning	+1	-1	+1	+1	-1
جو(do:)	0	0	0	0	0
بھی(b^hi:)	1	0	0	0	0
گفتگو(ft̪u:)	1	1	1	1	0
کر(kər)	1	1	1	1	0
اپنی(:pni:)	1	0	0	1	0
فرد(fərd)	0	1	0	0	0
اکثر(:ksər)	1	0	0	0	0
دمہ(dəmø)	0	0	1	0	0
مریض(mərz)	0	0	1	0	0
طویل(t̪əi:l)	0	0	1	0	0
باہر(ba:)	0	0	1	0	0
جا(ja:)	0	0	1	0	0
عمل(øəməl)	0	0	0	1	0
دارومدار(da:ro:məda:r)	0	0	0	1	0
استعمال(:st̪ma:l)	0	1	1	0	1
باہر(ba:)	0	0	1	0	0
طویل(t̪əi:l)	0	0	1	0	0
صحیح(sə)	0	0	0	1	1
تحریر(t̪e:rr)	0	0	0	0	1
متعلق(mt̪:lq)	0	0	0	0	1
ماہر(ma:r)	0	0	0	0	1

Table 2: Bag of Words Complete Sentence Model

We can observe that the feature vector created from the complete sentence model contains all the key words helping in sense disambiguation as well as the feature values of the sentences belonging to the same sense are now more similar to each other than before. Thus bag of words complete sentence model is outperforming bag of words limited size window model on these 5 sentences for one polysemous word. This motivates us to investigate the accuracies of the two data representations in this study.

4. Experiments

The sequence of our methodology is shown in Fig 1.

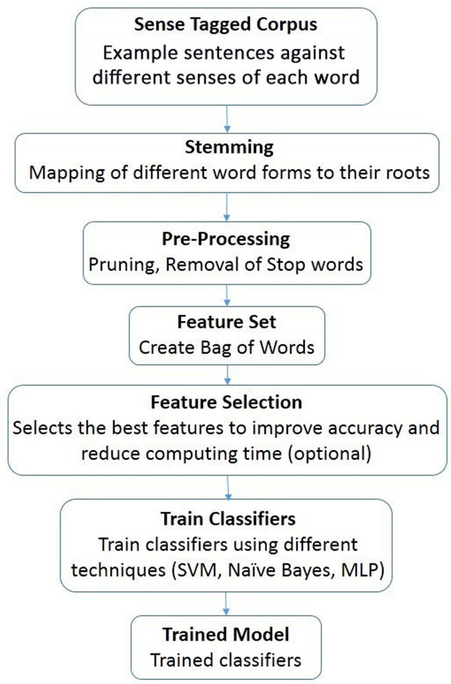


Figure 1: Overview of Supervised Machine Learning

We explain these steps in the following subsections.

4.1. Sense Tagged Corpus

A sense tagged corpus is a large collection of sentences with labeled senses of the polysemous words. We extracted data for our experiments from the sense-tagged Urdu CLE digest corpus [14] which consists of 100,000 words. We extracted those words from the corpus which had exactly 2 senses and more than 7 instances. Our final data consists of eleven words which are shown in table 3 alongwith the number of sentences for each word.

4.2. Stemming and Pre-Processing

After collecting the sentences, some pre-processing steps needed to be performed before any machine learning algorithms can be applied .We performed the following pre-processing steps.

1. Removal of Punctuation Marks: A list of all punctuation marks was maintained and a parser was used to parse all the sentences and remove their occurrences.
2. Stemming : Stemming means mapping different words to their roots. The stemming software created by the Center for Language Engineering for Urdu[16] was used for this purpose.
3. Removal of Stop Words : In order to remove all the stop words or closed class words from the data, we referred to the Urdu Closed Class Word List compiled by the Center for Language Engineering[16] and removed their occurrences from the data.
4. Further Cleaning : We removed any extra spaces as well as any unwanted characters for the purpose of generating a fully clean data set.

4.3. Feature Set and Feature Selection

From the remaining contents of the sentence, the feature vector is created according to the data representation model. In the case of the bag of words limited size window model, we tested several window sizes and found that a window size of 5 was giving the best results. Thus a bag of size 10 was created, containing 5 features to the left and 5 to the right of the target word, for each occurrence of the target word. If a target word occurred multiple times in a sentence then we created a bag of word for each instance. On the other hand for the bag of words with complete sentence model, multiple occurrences in a single sentence were not important and only one feature vector was created containing

Word	Meanings in English	No. of Sentences
اکثر(:ksər)	frequent	11
	majority	19
اندر(:ndər)	inside a person	15
	inside a place	26
انگریزی(:nr:z)	English (adjective)	14
	English (noun)	16
ایسا(e*sa:)	like	20
	such a	19
پاس(pa:s)	near	29
	possession	30
ترقی(tərəq)	progress	15
	promotion	14
خیال(k^{h}əja:l)	idea	21
	care	15
زبان(zəba:n)	language	15
	tongue	30
عمل(əməl)	act upon	21
	an act	14
کبھی(kəb^{h})	ever	15
	sometimes	25
کتاب(kta:b)	ordinary book	19
	divine book	15

Table 3: Words with Sense IDs and Number of Sentences

all the words in the sentence. Feature ranking was then applied on the feature vectors and the top 5, 10, 20, 50, 100, 150, 200, 250 features were selected. The feature vector for some words was small allowing selection of only upto top 150 features whereas some words had large feature vectors allowing selection of as much as top 250 features. Many feature ranking algorithms or metrics are available for text classification[29] from which we used |tpr-fpr| metric where tpr: true positive rate and fpr: false positive rate for selecting the top features in this study (Eq. 1). This step selects the most relevant or influencing features and removes the extraneous features that confuse the classifier.

$$F.Ranking = |tpr - fpr| \tag{1}$$

4.4. Classifiers

We used two classifiers in our experiments, the naïve bayes classifier [30] and the support vector machines (SVM) [30] because they have been used most extensively in text classification. We used a linear kernel with SVM because it has been the most widely used in this domain. We trained and tested the classifiers using the popular machine learning tool WEKA [31].

4.5. Performance Evaluation

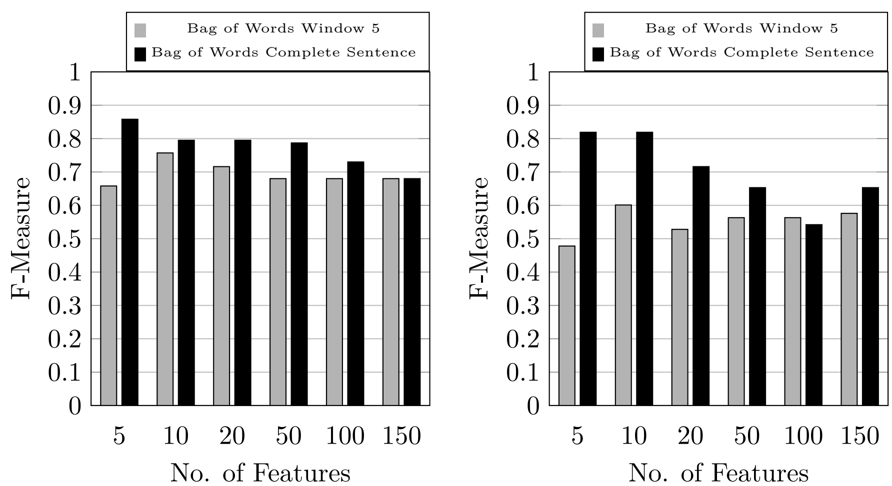
For evaluation purposes we used the leave-one-out cross fold validation (LOOCV) technique [32]. This technique takes 1 instance at a time for testing purposes and uses the rest of the instances for training. This process is repeated so that each instance has been treated as a testing element once.

For measuring the accuracy of our experimental results we used the F-Measure[29]. The F-Measure is calculated as $2*Precision*Recall/(Precision+Recall)$. The Recall is the proportion of those instances which were classified as a particular sense S among all instances that actually belong to that sense. The Precision is the proportion of all those instances which actually have the sense S among all those that are classified as S.

5. Results

The results of the experiments for each of the 11 words were recorded and analyzed. We provide a detailed description of the results for each word and show a bar-graph of the F-measure values for both representation models using both classifiers with all the feature vector sizes.

Figure 2 shows the graph of the results for the word اکثر(:ksər). It can be seen that the Bag of Words Complete Sentence model performs better than the Bag of Words Window size 5 model in the majority of the cases with the highest accuracy being 85.8% while using the top 5 features. Also, the Naive Bayes calssifier performs better than the support vector machines in all the cases.

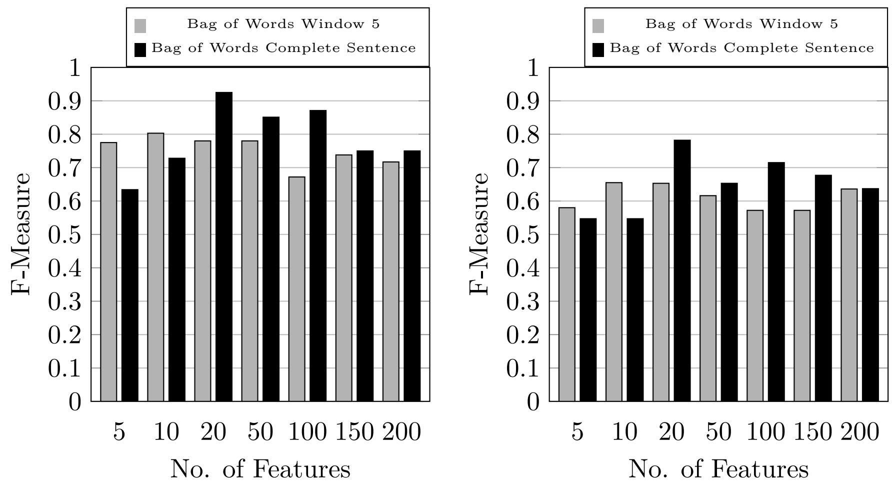


(a) Naive Bayes

(b) SVM

Figure 2: اکثر

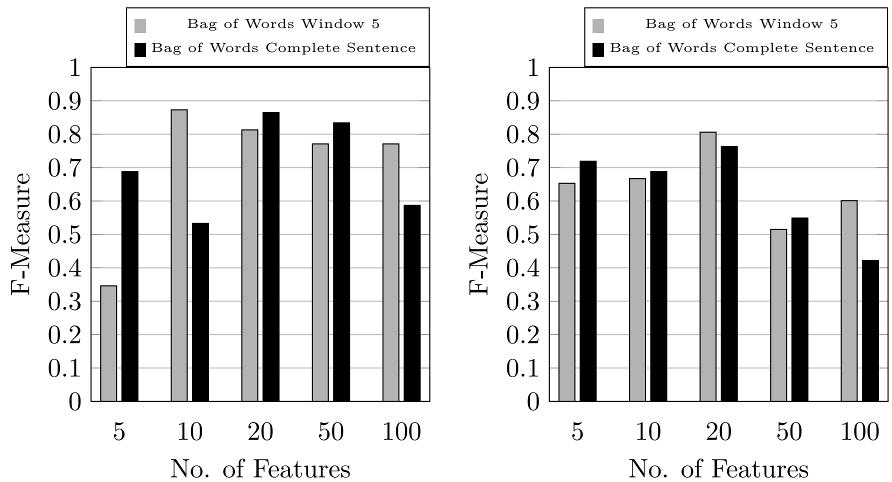
Figure 3 presents the results obtained for the word اندر(:ndər). Bag of words Complete sentence model again dominates the window size 5 model with the highest accuracy being 92.5% with the top top 20 features using Naive Bayes Classifier.



(a) Naive Bayes (b) SVM

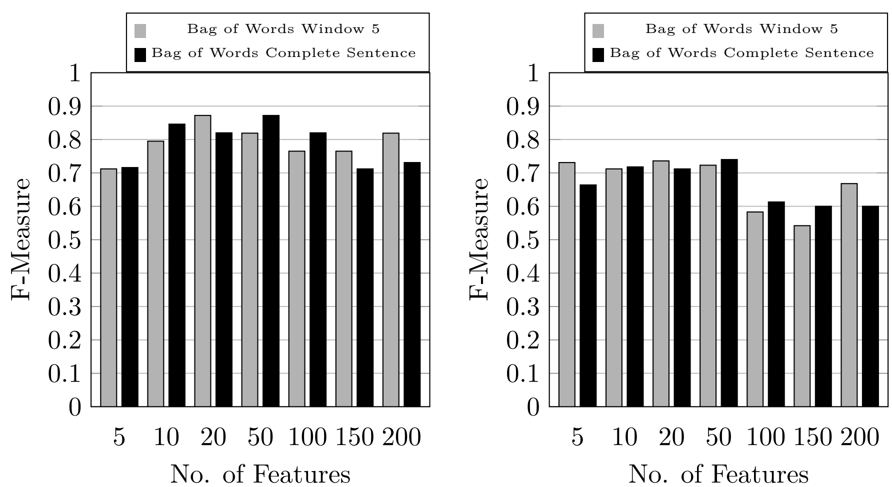
Figure 3: اندر

For the word انگریزی(:nr:z) the outcome of the various experiments are shown in Figure 4. Although the highest accuracy 87.3% is achieved using the window size 5 model, the complete sentence model has a higher total number of wins and thus shows better performance. There are also some instances where the Support Vector Machines perform better than Naive Bayes. These discrepancies from the norm can be due to insufficient training data.



(a) Naive Bayes (b) SVM

Figure 4: انگریزی

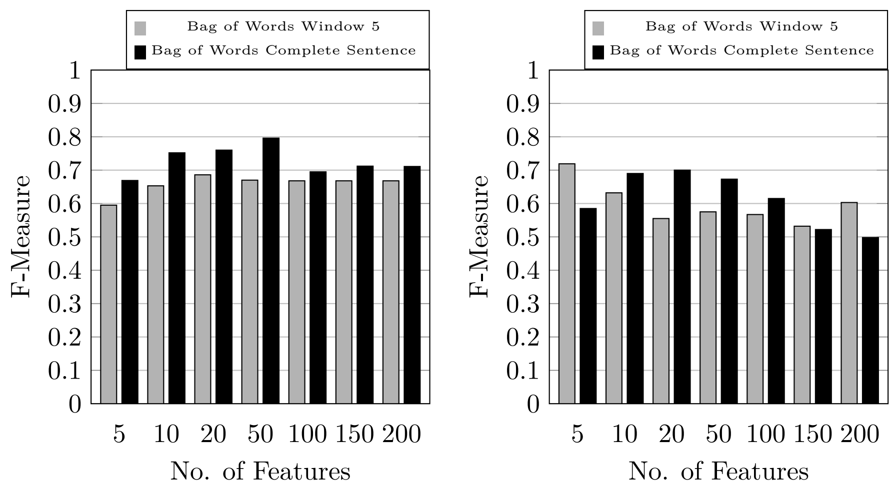


(a) Naive Bayes (b) SVM

Figure 5: ایسا

Figure 5 shows the results for the word ایسا(e:sa:). The two models are close in comparison but the bag of words complete sentence model again has a higher number of total wins. The highest accuracy 87.2% is achieved by both models

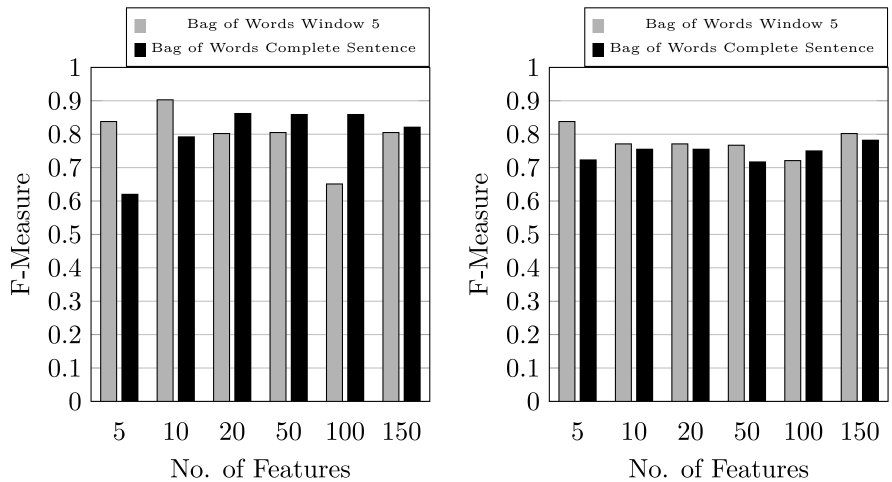
using Naive Bayes classifier.
Figure 6 presents the results for the word پاس(pa:s). The Bag of Words Complete Sentence model again outperforms the Window Size 5 model with the highest acccuracy of 79.6% while using the top 50 features. The Naive Bayes classifier dominates the SVM classifier in all the cases.



(a) Naive Bayes (b) SVM

Figure 6: پاس

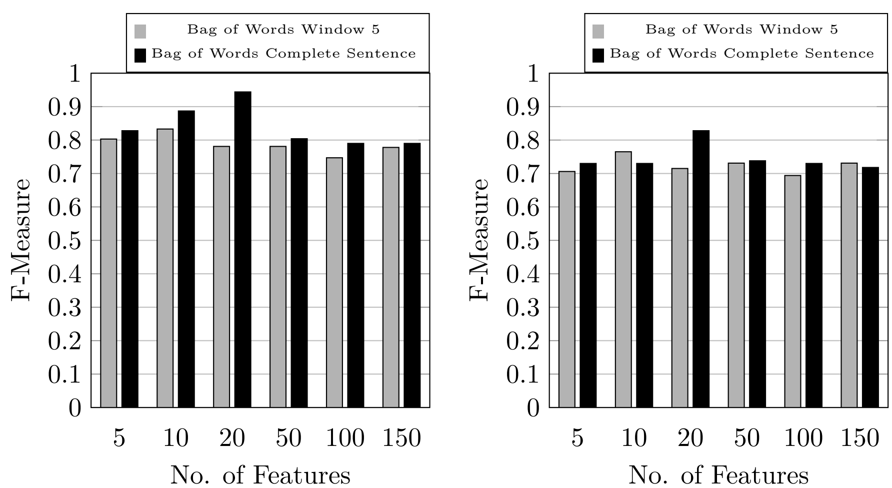
The findings for the word ترقی(tərəq) are presented in Figure 7. This word presented an interesting scenario where the Bag of Word Window Size 5 performed better than the complete sentence model with the highest accuracy being 90.3% with the top 10 features. Again, the anomaly in this result can be attributed to insufficient training data to capture all possible usages of the target word.



(a) Naive Bayes (b) SVM

Figure 7: ترقی

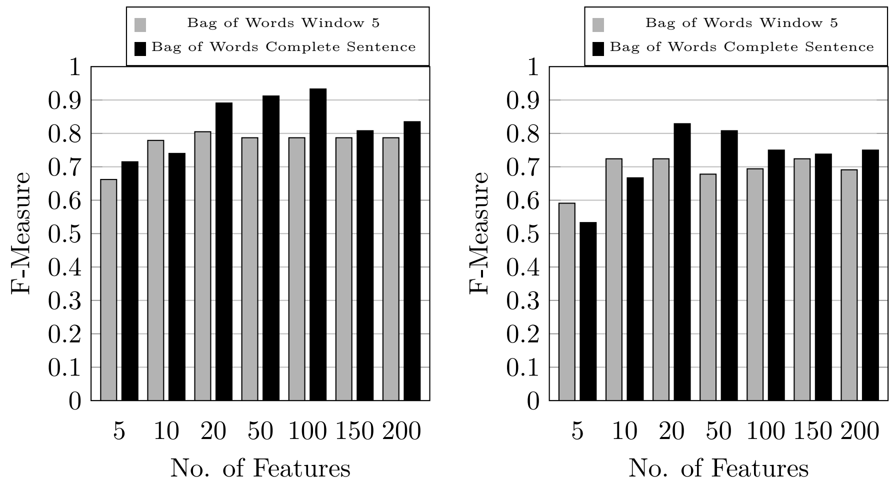
Figure 8 show the results for the word خیال(kʰəja:l). The Bag of Words Complete Sentence model again dominates with the best accuracy being 94.4% with the top 20 features using Naive Bayes classifier.



(a) Naive Bayes (b) SVM

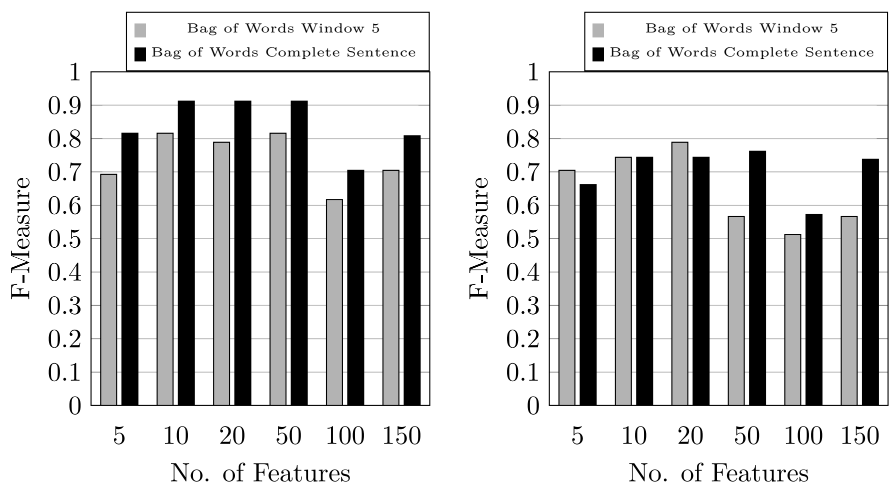
Figure 8: خیال

The output of the experiments on the word زبان(zəba:n) are shown in figure 9. The Bag of Words Complete Sentence model performs better in majority of the cases with the highest accuracy of 93.3% with the top 100 features.



(a) Naive Bayes (b) SVM

Figure 9: زبان

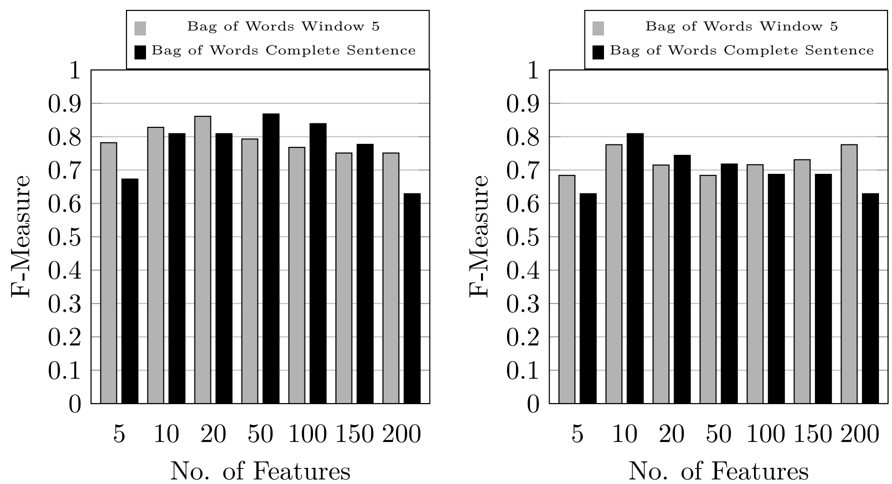


(a) Naive Bayes (b) SVM

Figure 10: عمل

The word عمل(əməl) gives the best results among all words (Figure 10) where the highest accuracy 91.2% is achieved by the Bag of Words Complete Sentence model by all top 20, 50 and 100 features using the Naive Bayes classifier.

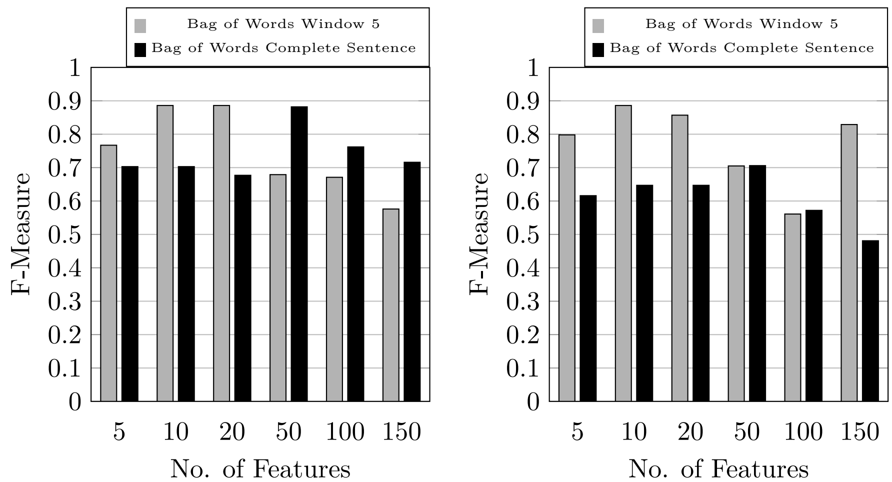
Figure 11 gives the results for the word کبھی(kəb^{h}). Although the greatest accuracy 86.8% is given by the Bag of Words Complete Sentence, the Bag of Words Window Size 5 model show overall better performance by winning in 8 out of 14 total cases.



(a) Naive Bayes (b) SVM

Figure 11: کبھی

The outcomes of the experiments on کتاب(kta:b) are given in figure 12. For this particular word, both the models gave excellent results, the Bag of Words Complete Sentence model gave 88.2% accuracy with the top 50 features and the Bag of Words Window Size 5 gave 88.6% accuracy with Support Vector Machine for top 10 and 20 features.



(a) Naive Bayes

(b) SVM

Figure 12: كتاب

6. Discussion

Table 4 shows a summary of the best results for each word with the corresponding data representation model and classifier. One clear observation is that the Naïve Bayes classifier shows better results than Support Vector Machines in almost all cases. This could be because the Naïve Bayes classifier assumes independence among the features, and the top ranked features for Urdu sentences are independent of each other.

As for the data representations, the Bag of Words Complete Sentence performs significantly better than the Bag of Words Window Size 5 model as depicted in Fig 13. This can be attributed to the fact that we are capturing more information in the complete sentence model. In Urdu language, the sentence structure is such that the decisive context words are often placed in distant corners of the sentence and the complete sentence model performs better than limited sentence model. We limited our study to words within the same sentence as we carried forward the assumption of a sentence being a significant determinant of the sense of a polysemous word.

The two words for which the Bag of Words with Window Size 5 model gave a better result than the Complete Sentence model could be due to the data being insufficient or biased such that the most indicative context words in the example sentences occurred within a window of 5 words from the target word. Expanding the data set might yield better results.

Word	F-Measure Bag of Words Window 5	Classifier	F-Measure Bag of Words Complete Sentence	Classifier
اکثر	0.757	Naive Bayes	0.858	Naive Bayes
اندر	0.803	Naive Bayes	0.925	Naive Bayes
انگریزی	0.873	Naive Bayes	0.865	Naive Bayes
ایسا	0.872	Naive Bayes	0.872	Naive Bayes
پاس	0.719	Support Vector Machine	0.796	Naive Bayes
ترقی	0.903	Naive Bayes	0.862	Naive Bayes
خیال	0.833	Naive Bayes	0.944	Naive Bayes
زبان	0.805	Naive Bayes	0.933	Naive Bayes
عمل	0.816	Naive Bayes	0.912	Naive Bayes
کبھی	0.861	Naive Bayes	0.868	Naive Bayes
کتاب	0.886	Naive Bayes + Support Vector Machine	0.882	Naive Bayes

Table 4: Summary

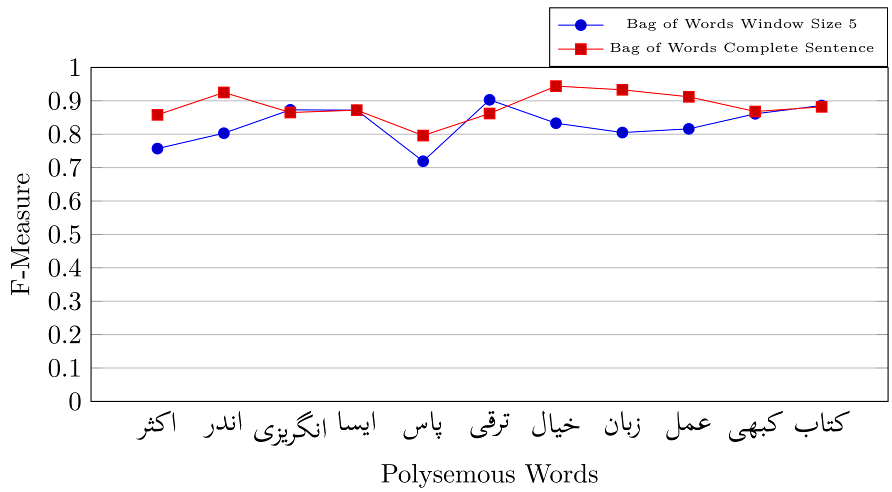


Figure 13: Comparison of the Accuracies of the Two Bag of Words Data Representation Models

7. Conclusion

In this study, we have taken eleven Urdu words which have two senses each and gathered example sentences against each sense from the CLE's Urdu digest corpus. We then investigated the effect of two different bag of words data representation techniques on word sense disambiguation in Urdu. We applied supervised machine learning using both Naive Bayes and Support Vector Machines classifiers on the respective data representation and found out that the Bag of Words complete sentence model completely dominates the Bag of Words limited window size model due to Urdu's unique sentence structure.

This work is a start towards the problem of word sense disambiguation for Urdu language and can be expanded to more words and senses. Several totally different feature sets that are being used in text classification and natural language processing can be applied. Moreover different classifiers and data representations can be tried. Another interesting research would be to investigate the use of semi supervised learning and bootstrapping to enhance the Sense Tagged Corpus and try to improve on the amount of data for Urdu word sense disambiguation as well as the accuracy. Similarly unsupervised techniques can be used to find out if more senses of a given word exist in a corpus. Therefore the field is wide open for further research and improvements in Urdu language.

References

[1] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. J. Miller, Introduction to wordnet: An on-line lexical database*, International journal of lexicography 3 (4) (1990) 235–244.

[2] M. H. Elyasir, K. Sonai Muthu Anbananthen, et al., Comparison between bag of words and word sense disambiguation, in: 2013 International Conference on Advanced Computer Science and Electronics Information (ICACSEI 2013), Atlantis Press, 2013.

[3] D. Vickrey, L. Biewald, M. Teyssier, D. Koller, Word-sense disambiguation for machine translation, in: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2005, pp. 771–778.

[4] J. C. Hung, C.-S. Wang, C.-Y. Yang, M.-S. Chiu, G. Yee, Applying word sense disambiguation to question answering system for e-learning, in: Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on, Vol. 1, IEEE, 2005, pp. 157–162.

[5] C. Stokoe, M. P. Oakes, J. Tait, Word sense disambiguation in information retrieval revisited, in: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, ACM, 2003, pp. 159–166.

[6] J. H. Martin, D. Jurafsky, Speech and language processing, International Edition.

[7] D. Martínez, et al., Supervised corpus-based methods for wsd, in: Word Sense Disambiguation, Springer, 2006, pp. 167–216.

[8] T. Pedersen, Unsupervised corpus-based methods for wsd, Word sense disambiguation: algorithms and applications (2006) 133–166.

[9] R. Mihalcea, Knowledge-based methods for wsd, Word Sense Disambiguation: Algorithms and Applications (2006) 107–131.

[10] H. T. Ng, Getting serious about word sense disambiguation, in: Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How, 1997, pp. 1–7.

[11] R. Mihalcea, Semcor semantically tagged corpus, Unpublished manuscript.

[12] F. Bond, T. Baldwin, R. Fothergill, K. Uchimoto, Japanese semcor: A sense-tagged corpus of japanese, in: Proceedings of the 6th Global WordNet Conference (GWC 2012), 2012, pp. 56–63.

[13] Y. Wu, P. Jin, Y. Zhang, S. Yu, A chinese corpus with word sense annotation, in: Computer Processing of Oriental Languages. Beyond the Orient: The Research Challenges Ahead, Springer, 2006, pp. 414–421.

[14] S. Urooj, S. Shams, S. Hussain, F. Adeeba, Sense tagged cle urdu digest corpus, in: Proc. Conf. on Language and Technology, Karachi, 2014.

[15] A. Naseer, S. Hussain, Supervised word sense disambiguation for urdu using bayesian classification.

[16] Center for language engineering, al-khawarizmi institute of computer sciences, university of engineering & technology, lahore.
URL `http://www.cle.org.pk/`

[17] S. Urooj, S. Hussain, F. Adeeba, F. Jabeen, R. Parveen, Cle urdu digest corpus, LANGUAGE & TECHNOLOGY (2012) 47.

[18] R. Mihalcea, Using wikipedia for automatic word sense disambiguation., in: HLT-NAACL, 2007, pp. 196–203.

[19] H. T. Ng, H. B. Lee, Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach, in: Proceedings of the 34th annual meeting on Association for Computational Linguistics, Association for Computational Linguistics, 1996, pp. 40–47.

[20] Y. K. Lee, H. T. Ng, An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation, in: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, Association for Computational Linguistics, 2002, pp. 41–48.

[21] T. Pedersen, A simple approach to building ensembles of naive bayesian classifiers for word sense disambiguation, in: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, Association for Computational Linguistics, 2000, pp. 63–69.

[22] J.-B. Gao, B.-W. Zhang, X.-H. Chen, A wordnet-based semantic similarity measurement combining edge-counting and information content theory, Engineering Applications of Artificial Intelligence 39 (2015) 80–88.

[23] H. Liu, V. Teller, C. Friedman, A multi-aspect comparison study of supervised word sense disambiguation, Journal of the American Medical Informatics Association 11 (4) (2004) 320–331.

[24] S. Singh, T. J. Siddiqui, Role of semantic relations in hindi word sense disambiguation, Procedia Computer Science 46 (2015) 240–248.

[25] S.-J. Ker, J.-N. Chen, Adaptive word sense tagging on chinese corpus, in: PACLIC, Vol. 18, Citeseer, 2004, pp. 8–10.

[26] C. A. Le, A. Shimazu, High wsd accuracy using naive bayesian classifier with rich features, in: Proceedings of PACLIC, Vol. 18, 2004, pp. 105–113.

[27] M. Hamidi, A. Borji, S. S. Ghidary, Persian word sense disambiguation, in: Proceeding of 15-th Iranian Conference of Electrical and Electronics Engineers (ICEE 2007), Tehran, Vol. 25, 2007.

[28] [link].
URL https://en.wikibooks.org/wiki/Urdu/Alphabet

[29] G. Forman, An extensive empirical study of feature selection metrics for text classification, The Journal of machine learning research 3 (2003) 1289–1305.

[30] E. Alpaydin, Introduction to machine learning, MIT press, 2014.

[31] M. H. R. K. P. R. A. S. R. R. Bouckaert, E. Frank, D. Scuse, Weka manual for version 3-7-12.

[32] T. M. Mitchell, Machine learning. wcb (1997).