TÜBİTAK

Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 4391 – 4405
© TÜBİTAK
doi:10.3906/elk-1408-77

Exploring feature sets for Turkish word sense disambiguation

Bahar İLGEN[1,2,*], Eşref ADALI[1], Ahmet Cüneyd TANTUĞ[1]
[1]Computer and Informatics Faculty, İstanbul Technical University, İstanbul, Turkey
[2]Computer Engineering Department, İstanbul Kültür University, İstanbul, Turkey

Received: 12.08.2014 • **Accepted/Published Online:** 07.08.2015 • **Final Version:** 20.06.2016

Abstract: This paper presents an exploration and evaluation of a diverse set of features that influence word-sense disambiguation (WSD) performance. WSD has the potential to improve many natural language processing (NLP) tasks as being one of the most crucial steps in the area. It is known that exploiting effective features and removing redundant ones help improving the results. There are two groups of feature sets to disambiguate senses and select the most appropriate ones among a set of candidates: collocational and bag-of-words (BoW) features. We introduce the effects of using these two feature sets on the Turkish Lexical Sample Dataset (TLSD), which comprises the most ambiguous verb and noun samples. In addition to our results, joint setting of feature groups has been applied to measure additional improvement in the results. Our results suggest that joint setting of features improves accuracy up to 7%. The effective window size of the ambiguous words has been determined for noun and verb sets. Additionally, the suggested feature set has been investigated on a different corpus that had been used in the previous studies on Turkish WSD. The results of the experiments to investigate diverse morphological groups show that word root and the case marker are significant features to disambiguate senses.

Key words: Bag-of-words features, collocational features, feature selection, supervised methods, word-sense disambiguation

1. Introduction

The identification of word meanings is required in almost all applications of the natural language processing (NLP) area to provide them proper functioning. These applications include the areas such as information retrieval (IR), information extraction (IE), machine translation (MT), semantic annotation (SE), question answering (QA), and many others. Basically, the aim of a word-sense disambiguation (WSD) system is to automatically determine the most probable sense of a polysemous word among the possible set of sense candidates in a given context. In the following example, the meaning of “bass” is a kind of fish. “I went fishing for some sea bass”. One can understand that bass is a fish but a computer will find several meanings such as 1) low frequency, 2) bass guitar, 3) fish. The WSD process solves sense disambiguation and finds the most appropriate solution.

There are four conventional approaches for WSD consisting of (1) dictionary- and knowledge-based methods, (2) supervised methods, (3) unsupervised methods, and (4) semisupervised (or minimally supervised) methods. The family of knowledge-based methods primarily relies on dictionaries, thesauri, ontologies, and lexical knowledge bases. External knowledge sources may be structured or unstructured such as corpora and

[*]Correspondence: b.ilgen@iku.edu.tr

web resources [1]. Unsupervised methods utilize external information and work on raw unannotated corpora, while supervised methods use sense-annotated corpora to train from. Recently, minimally supervised methods have also gained attention since the sense annotation scheme is labor intensive and expensive.

A secondary classification for generic WSD can be made by considering two variants: (1) lexical sample (LS) task and (2) all-words (AW) task. The former approach disambiguates the occurrences of a small sample of a target word that has been determined previously. Since the words and the set of senses are limited, supervised machine learning (ML) methods are usually used to handle LS tasks. In contrast, AW approaches disambiguate all the words in a running text. All the entries in a given WSD system are required to be disambiguated. AW task and part of speech (POS) tagging are similar except that the former requires a much larger set of tags [2].

Knowledge is the essential component of a WSD system that can be acquired from dictionaries or learned from a training corpus. The sources are classified into "lexical knowledge" (e.g., sense frequency, concept trees, selectional restrictions, subject code, and the POS information) and "learned world knowledge" (e.g., indicative words, syntactic features, domain specific knowledge, and parallel corpora) categories [3]. It is usually observed that the unsupervised systems benefit from lexical knowledge sources, while supervised systems use world knowledge. However, in practice combinations of these sources have been used in WSD systems.

ML techniques are used to automatically acquire disambiguation knowledge in the scope of corpus-based WSD methods. A typical WSD system may utilize sense-tagged corpora, online dictionaries, and large-scale linguistic resources. Many NLP systems rely on linguistic knowledge acquired from hand-labeled training corpus and ML methods [4]. The supervised methods of the WSD are classified according to the induction principle they use to acquire model or rules. These are probabilistic models (e.g., naïve Bayes), similarity based methods (e.g., k-nearest neighbor – kNN algorithm), methods based on discursive properties (e.g., one sense per discourse/collocation, attribute redundancy), methods of discriminative rules (e.g., decision lists, decision trees, or methods based on rule combination), linear classifiers, and kernel-based methods. Before applying the ML algorithm, all the examples of a particular ambiguous word have to be encoded in a way the learning algorithm can handle.

Compared to the other subjects in NLP such as POS determination and syntax parsing, the WSD problem introduces extra difficulties. Since each word is associated with a unique meaning, the complete training set requires a huge number of examples. This case is explained by the fact that word meanings in natural languages are distributed by Zipf's law [5]. The problem of language sparsity can be dealt with by selecting features used in training algorithms. These features can be found in a local or wider context.

Collocational and bag-of-words (BoW) features are two important classes that are generally extracted from neighboring contexts. Almost all of these approaches are employed by defining a window of "n" content words around the word to be disambiguated. Collocational features encode the information about the lexical neighbors of specific positions located to the left or right of the ambiguous word. The basic elements may consist of the word, its root form, and the part of speech information. The following sentence example in Figure 1 shows the ambiguous word "bass" and the surrounding information around it. The abbreviations of "pre" and "fol" stand for the previous and following words, respectively. The optimal extent of the window around the ambiguous target word has been focused on in several studies [6]. BoW is the second feature set in which the text is treated as an unordered bag of words. Similarity measures of the BoW approach are calculated by looking at the semantic similarity between all the words in the window regardless of their positions.

An electric guitar and ***bass*** *player stand off to one side, not really part of the scene just as a sort of nod to gringo expectations perhaps.*

pre2-word: guitar, pre2-pos: NN1
pre1-word: and, pre1-pos: CJC
fol1-word : player, fol1-pos: NN1
fol2-word : stand, fol2-pos: VVB

Figure 1. An example of a windowing scheme in WSD.

Especially for English many algorithms and methods have been developed for WSD. Because Turkish is an agglutinative language and requires extensive usage of suffixes, the results of previously developed methods and algorithms cannot be directly used for Turkish. For languages that have agglutinative morphology, it is quite possible to produce thousands of forms for a given root word. In this study we focus on effective features of Turkish, which has completely different characteristics such as syntactic structure and suffixes. Natural languages also differ in polysemy degree and the high polysemy degree of Turkish is another reason for us to investigate effective features.

The rest of the paper is organized as follows. In Section 2, related work is summarized. Section 3 describes the datasets that we used in the experiments. Section 4 introduces the properties of the collocational and BoW features. Section 5 presents the accuracy results of the selected feature sets. This section also covers the accuracy results for joint setting of feature sets and additional experiments on effective window size. Finally, Section 6 contains the conclusion.

2. Related work

Feature selection is a crucial step of WSD systems in supervised training that has direct effect on system performance. As mentioned in Section 1, there are several knowledge sources to remove lexical ambiguity. The impact of using predictive features and removing redundant ones has been investigated in many studies [7,8].

The features have been divided into topical and local contextual features in [9]. Topical features indicate the presence of key words occurring anywhere in the sentence. The surrounding sentences are usually taken as context. Local features comprise the collocational features and require no linguistic preprocessing beyond POS tagging, and syntactic and semantic features. The difficulties in sense tagging tasks of Chinese and English verbs have been compared in [10] by using similar features. The useful contextual features have been investigated for both languages. Their results suggest that richer linguistic features are beneficial for English WSD. Additionally these features may not be as useful for Chinese WSD.

In [11], the main feature types have been grouped into three main sets. These are local collocations, syntactic dependencies, and global features. In total six feature sets are used: bag-of-words, local collocations, syntactic dependencies, bag-of-bigrams, all features except bag-of-words, and all features. In order to organize experiments, different editions of Senseval[1] have been used. They used Senseval-2 Lexical Sample data, which is a sample of BNC [12], mostly for development and tuning of the parameters. Senseval-3 LS and AW datasets have been used for testing. They reported that the AW set is the best single classifier in every method except one. Taking into account each feature set separately, it is stated that local collocational features discriminate better than BoW features.

Several combinations of features have been tested for Turkish in [13] and [7]. Some of the features are root words, POS information, word possessors, and case markers. A study of feature selection in a supervised learning method of WSD is presented in [14]. The features in the study are automatically defined using a

[1] http://www.senseval.org

function. Some of the features of the work are 0-features, S-features, Q-features, and Km-features. These features represent the target word; words in positions ± 1, ± 2, and ± 3; POS tags of words in positions ± 1, ± 2, and ± 3; and lemmas of nouns in context at any position (occurring at least m% with a sense), respectively. The paper shows that not all the words are better disambiguated by using the same feature set.

In [15] an instance-based WSD system has been investigated. The approach integrates the diverse set of knowledge sources for disambiguation. The extracted features of the study are POS and morphological form, unordered set of surrounding words, local collocations, and verb–object syntactic relation. It is stated in the study that the approach achieves higher accuracy than previous work.

There are also works [16,17] in which unordered sets of surrounding words have been used to perform WSD. In [18], surrounding words, POS, and morphological forms have been used. The use of general-purpose inductive logic programming (ILP) has been examined to construct a set of features using semantic, syntactic, and lexical information in [19]. They reported that the use of ILP with diverse knowledge sources provides improvement in the WSD field. The approach in [20] combines various sources of knowledge through combinations of two WSD methods. Scott and Matwin [21] examined the alternative ways to represent text features based on semantic and syntactic relations between words (phrases, synonyms, and hypernyms).

Chodorow et al. [22] consider four types of contextual features: (1) topical cues of open class words (nouns, verbs, adjectives, adverbs), (2) local open class words found in the narrow context of the target word, (3) local closed class items (nonopen class words such as prepositions and determiners), (4) local part-of-speech tags.

There are also several works [23] on determining window size since windowing schemes are needed in NLP-related tasks. It is important to know the effective distance around the target word in advance. In [24], the performance of different windowing schemes using two conceptual hierarchies-based semantic similarity metrics was analyzed. The maximum relatedness disambiguation algorithm has been used for experiments [25] The algorithm uses a quantitative measure of similarity between word senses in context to disambiguate the ambiguous word. Similar experiments have been carried out on different windowing schemes. Experiments have been done by using WordNet [26] and a subset of nouns in the SenSeval-2 English LS dataset. The subset they used contains 914 noun instances of the data source for 5 target words (art, authority, bar, bum, and chair). Different similarity metrics have been tested for varying windowing schemes. Their results suggest that the best performing window size on average is 7.

Yarowsky [27] has conducted several experiments using varying window sizes. The findings suggest that local ambiguities need smaller windows (window of k = 3 or 4). On the other hand, semantic or topic-based ambiguities need larger windows of 20 to 50 words.

3. Dataset

3.1. Turkish Lexical Sample Dataset

In the scope of this work, we have utilized two corpora. The Turkish Lexical Sample Dataset (TLSD) is the first one that we used in the experiments to determine the effective features and window size. This lexical sample data has been built up to conduct our previous studies on Turkish [28].

Initially we determined the candidate words for Turkish noun and verb sets to be disambiguated. These words were chosen as candidates among the highly ambiguous words of the language, assuming that a method performing well on these highly ambiguous words would also perform well on other words. As a result of our simple analysis on the dictionary of the Turkish Language Association (TLA) [29], we found the following numbers for sense ambiguities. The number of distinct lemmas in the dictionary is 68,639 and the average number of senses per lemma is 1.61. However, 51,958 of these words have only one sense, and so they are out of the scope of the WSD task. After exclusion of these words that have only one sense in the dictionary, the total number of lemmas has become 16,681, and the corresponding average number of senses per lemma is calculated as 3.53. After analysis of candidate ambiguous words, we determined highly ambiguous noun and verb sets.

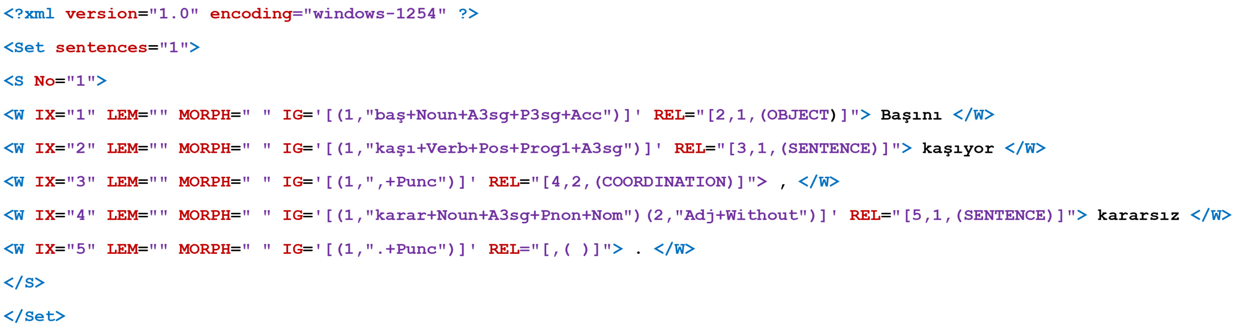
The TLSD comprises ambiguous words of noun and verb sets each of which has 15 candidates. It includes at least 100 samples for each chosen ambiguous word. Well-known Turkish websites on news, health, education, and sports were taken as the main resources for the corpus. In the samples, the ambiguous word that will be disambiguated is marked as headword. We follow the "one-sense per sample" principle and so each sample text includes only one sense of the headword. According to the context, 5 human annotators labeled these words with the proper senses in the dictionary of the TLA [29]. The average polysemy degrees of the noun and verb sets have been calculated as 10.67 and 26.53, respectively. As an example from the noun group, the Turkish word "baş" ("head" in English) has 13 different senses. Some of the senses can be listed as: (1) head of a person, (2) leadership, (3) beginning, (4) highest point of a geographic area, (5) basic etc. Moreover, some of the verbs may have up to 40 ˜50 senses in Turkish. For example "çıkmak" (exit, quit) has 56 different meanings in the dictionary of the TLA (http://www.tdk.gov.tr/). It should be noted that this property of Turkish introduces extra difficulties in the WSD task.

3.2. METU-Sabancı Turkish Treebank

As the second corpus, we used the METU-Sabancı Turkish Treebank [30,31] in order to compare our results with the previous work [32] using the same corpus. Comparing our results with the findings of the previous work is important since there are very few works on WSD that conduct similar experiments on Turkish noun and verb sets.

The METU Turkish corpus is available for academic purposes. It has two parts: the main corpus and the Treebank portion. In the second part of the study, we repeated our experiments on the METU-Sabancı Turkish Treebank corpus, which has different ambiguous Turkish noun and verb groups. The sentences in the METU-Sabancı Turkish Treebank were prepared in XML format and provide syntactic features that can be utilized in the disambiguation process. The texts in the main part have been collected from different sources of written Turkish resources (books, papers, and newspapers) published in 1990 and afterwards. There are approximately two million words in the corpus. XML and TEI (Text Encoding Initiative) style annotation has been used to build a corpus similar to the BNC (British national corpus). The Treebank portion of the corpus has been built from 6930 sentences of the main part. The chosen sentences have been parsed, morphologically analyzed, and disambiguated. The distribution of the Treebank is similar to the main corpus. As an annotated resource, the METU-Sabancı Turkish Treebank corpus [30,31] has been used in previous studies [7,13,32]. Figure 2 displays a sample from the corpus.

One of the differences between these two corpora is the scope of the headword occurrences. Although the sentence level information can be gathered from Treebank, headword occurrences are available in paragraph scope in the TLSD.



```
<?xml version="1.0" encoding="windows-1254" ?>
<Set sentences="1">
<S No="1">
<W IX="1" LEM="" MORPH=" " IG='[(1,"baş+Noun+A3sg+P3sg+Acc")]' REL="[2,1,(OBJECT)]"> Başını </W>
<W IX="2" LEM="" MORPH=" " IG='[(1,"kaşı+Verb+Pos+Prog1+A3sg")]' REL="[3,1,(SENTENCE)]"> kaşıyor </W>
<W IX="3" LEM="" MORPH=" " IG='[(1,",+Punc")]' REL="[4,2,(COORDINATION)]"> , </W>
<W IX="4" LEM="" MORPH=" " IG='[(1,"karar+Noun+A3sg+Pnon+Nom")(2,"Adj+Without")]' REL="[5,1,(SENTENCE)]"> kararsız </W>
<W IX="5" LEM="" MORPH=" " IG='[(1,".+Punc")]' REL="[,( )]"> . </W>
</S>
</Set>
```

Figure 2. XML file sample of the METU-Sabancı Turkish Treebank Corpus.

4. Features

4.1. Collocational features

Collocational features encode the information about neighbors of the target word at the left or right positions. The words in the window, their root forms, and the POS information are the typical features at encoding grammatical local lexical features. These features can be extracted from text that has been segmented into POS tagged words. We have used the following information to extract features: (1) The target word, (2) The POS tag of the target word, (3) The words (if any) within ± 4 positions of the target word, (4) The POS of the words within ± 4 positions of the target word.

Although these effective features are common for all languages, some features may be language specific. As a member of the agglutinative languages family, Turkish is based on suffixation, which distinguishes it from the majority of European languages. Grammatical functions in Turkish are indicated by adding suffixes to the stems. As a result, the number of generated POS features may be excessive in Turkish. In this study, we adjusted the window size by considering four words on the left side and four words on the right side of the target word.

Because of the agglutinative property with productive inflectional and derivational suffixations of Turkish, a preprocessing step is required to obtain root words and other morphological information. We used a finite-state two level Turkish morphological analyzer [33] for morphological decomposition. Since the output of this analyzer is ambiguous, a morphological disambiguation tool [34] is also utilized. An example output that we obtain after applying the morphological analyzer and disambiguation tool is shown in Figure 3. It is an example from the TLSD that shows the POS tagged version of the ambiguous word "baş" (head) together with the words that are placed in the ± 4 neighborhood. The words are shown in their root forms on the left side with their POS tags. In Turkish, it is common for a single word to have many features after the morphological analysis process. These features consist of major types (nouns, verbs, adjectives etc.), minor subparts, nominal forms, compounding/modality tags, polarity tags, tenses, or semantic markers [8].

It is clear from the Figure 3 that each root may have a different number of morphological features. Although it is not observed in Figure 3, some of the morphological features may contain derivational boundaries (DBs) that change the basic structure of the major root.

In the representation of Turkish morphology, the information encoded in complex agglutinative word structures is represented as a sequence of inflectional groups (IGs) separated by DBs [30]. The list of morphological categories (and a limited part of subcategories) used in the encoding of about 9000 possible IGs is given in Table 1. Detailed information about the Turkish morphological categories can be found in [30]. As an example in Figure 4, the word "kuvvetlendirme" (to make something become strong) has three DBs that change the category of the word. Both the category of the root and the final categories are noun. On the other hand, there are two intermediate categories as verb. For the experiments that we carried out, we took the only last tagset generated after DB and ignored the previous information.

kriz:(Noun)(A3pl)(Pnon) (Nom)
sonra:(Noun)(Zero)(A3sg)(P3sg)(Loc)
büyük:(Adj)
şirket:(Noun)(A3pl)(Pnon)(Gen)
<HEAD SENSE='baş' - SENSE_TDK_NO="2" ...
baş:(Noun)(A3sg)(P3sg)(Loc)
</HEAD>
bulun:(Adj)(PresPart)
yönetici:(Noun)(A3pl)(Pnon)(Gen)
görev:(Noun)(A3sg)(Pnon)(Nom)
değişim:(Noun)(A3pl)(P3sg)(Nom)

Figure 3. Window scope of the collocational features.

Table 1. Tagsets with corresponding subfeatures.

Main POS tags	Subtags
Verb	Able, Fut, Past, Acc, Caus, Pres, Neces, Pos, Pass, Neg, Zero, Prog1, Prog2, Narr, Cond, Axxx, Pxxx
Noun	Ness, Nom, Loc, Dat, Abl, Gen, Prop, Pnon, Ins, Dim, Zero, Acc, PastPart, FutPart, Agt, Axxx, Pxxx
Postp	PCNom, PCDat, PCAbl, PCIns
Adverb	AsLongAs, While, ByDoingSo, AfterDoingSo
Pron	Demons, Nom, Reflex, Pers, Dat, Pnon, Ques, Quant, Acc
Adj	With, PresPart, Without, Rel, PastPart, FutPart, FitFor
Ques	Pres, Past, Axxx
Num	Card, Ord, Range, Real
Punc/Dup/Interj/ Conj/Det	–No subfeature–

Kuvvet+len+dir+me
kuvvet+Noun+A3sg+Pnon+Nom^**DB**
+Verb+Acquire^**DB**
+Verb+Caus+Pos^**DB**
+Noun+Inf2+A3sg+Pnon+Nom

to make (something) become strong /
to strengthen (something)

Figure 4. Morphological analysis of "kuvvetlendirme".

In this study, we considered 119 features in total for each word in the window scope. Since the secondary features depend on their major roots, we determined succeeding subproperties for each major root, and these 119 features were organized under the label of major root categories such as "Verb", "Noun", "Adj", and "Pron". For example, major root "Num" has subfeatures such as "Card", "Ord", "Range", and "Real" while some of the subfeatures of the major root "Verb" consist of "Able", "Pos", "Fut", "Past", "Cond", etc. Since the window size varies around the headword in ± 4 range, the size of the whole feature vector became (119 × 9) features + sense-label + roots, which equals 1081. In Table 1, the main tags and some of the corresponding subtags that can be generated by the Turkish morphological analyzer [33] are shown. It can be seen from the table that some of the categories such as "Det", "Conj", and "Punc" (determiner, conjunction, punctuation) do not have subcategories.

Our previous work on Turkish summarizes the effective collocational features when window size is taken as ± 4 [8]. This work presents the effective features for ambiguous noun and verb sets. We took the elements between w_{-4} and $w_{+4}(w_{-4}, w_{-3}, w_{-2}, w_{-1}, h_0, w_{+1}, w_{+2}, w_{+3}, w_{+4})$ into consideration and obtained the efficient features among POS features and word stems.

In the scope of [8], the effective features among complete set of features for noun and verb sets have been determined separately. The correlation feature selection (CFS) has been applied to all samples in the TLSD to obtain the most common features of all ambiguous words in a set. The results for both sets show that the effective features are mostly located between w_{-2} and w_{+2}.

As an additional experiment, we have investigated the impact of three morphological groups on Turkish verb and noun sets. The inflectional markers that we considered are: (1) number/person agreement, (2) possessive agreement, and (3) case. Nominal forms (nouns, derived nouns, pronouns, participles, and infinitives) get these additional inflectional markers. The results have been obtained by using different combinations of these sets.

4.2. Bag-of-words features

The second type of feature set comprises the BoW information of neighbors around the target word. The BoW model refers to an unordered set of words, with their exact location ignored. All grammar, word relations, and even word order are disregarded within this model. The words in this model serve as features. The value of any feature is determined by counting the number of times the words occur in the region surrounding the ambiguous target word. This region is generally a fixed window with the ambiguous word as center. In the scope of the experiments, the following steps have been followed to apply the BoW approach:

- The TLSD has been morphologically analyzed and the ambiguity of the output has been removed.
- Stem forms of the words have been obtained.
- Most frequent "n" words of the data set have been determined.
- Both the training and test portions of the data set have been encoded using "n" most frequent words (i.e. each word in samples has been encoded considering the existence of the most frequent words in the sample).
- The algorithms that we utilized for collocational features have been used with bag-of-words features.

We did additional experiments to determine the effective number of BoW features initially and determined the most frequent content words. We scanned the entire training portion of each ambiguous word (i.e. we took 67% of the dataset as the training portion, which approximately equals 67˜70 samples in the dataset) and ranked the most frequent content words. We repeated this step 4 times and determined the most frequent 100, 75, 50, and 25 content words of the lexical samples. For each feature set, both the training and test portions of the dataset are encoded using this information. For example, if the most frequent 25 words are taken as features, both the training and test parts (samples) are encoded by using these words. We used a vector that is initialized by assigning “0” to each cell. Then the value of the cell is incremented by “1” if the feature exists in the lexical sample. Figure 5 shows the examples of vector structure that we used in the experiments. The cells of the vector indicate the existence of any feature in the context. The last cell is reserved for the sense label. After repeating similar tests for the 50, 75, and 100 most frequent words, we obtained the accuracy results by using the Weka [35] tool. Our findings show that the most frequent 75 and 100 content words yielded better accuracy results for verb and noun groups, respectively. In the rest of the experiments, we kept and used these settings to determine effective window size.

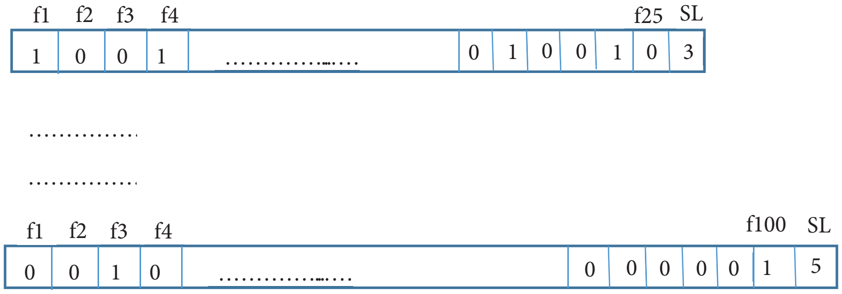


Figure 5. Bag-of-words feature vectors.

5. Experimental results

5.1. Determination of the effective window size

After choosing the number of optimal features for the BoW approach, we conducted experiments to determine the best window size. We took ± n words around the headword and decided to carry out our experiments for varying values of the “n”. Then the experiments were conducted for varying window size (± 30, ± 15, ± 10, and ± 5) of the preceding and following words.

The experiments have been repeated on two sets consisting of Turkish nouns and verbs of the TLSD. We followed the 2/3 splitting strategy and used the training portion to extract features. After determining the features to be used, ML algorithms have been investigated on the annotated dataset. In the scope of this work, we have tested four ML algorithms consisting of naïve Bayes, IBk (k-nearest neighbor), SVM (support vector machines), J48 (C4.5 algorithm-decision tree), and functional trees (FT). Weka 3.6.5 [35] has been used for the experiments. We used the default parameters in Weka for SVM classification: polynomial kernel and C = 1 for all the experiments.

There are some cases in which we have obtained better results using different windowing schemes. On the other hand, the average success ratio of naïve Bayes and FT algorithm is better for both the noun and verb groups. Although we selected these methods to present our results of the windowing scheme, other algorithms give similar and good results for varying window sizes. It is observed that the best results of the two groups are 65.8% and 56% points above the most frequents sense baseline of verbs and nouns, respectively [36]. The results show that the best window size for noun and verbs sets is 5.

Figure 6(a) presents the naïve Bayes and the FT algorithm results of the noun group for varying window sizes, respectively. Similarly, Figure 6(b) shows the accuracy results of the verb group. The average accuracy results of varying window size (WS) values have been presented together with the value of most-frequent-sense baseline (MFB). Because Turkish verbs are much more ambiguous than the noun group (i.e. sense labels up to 50 or more), the baseline values are lower.

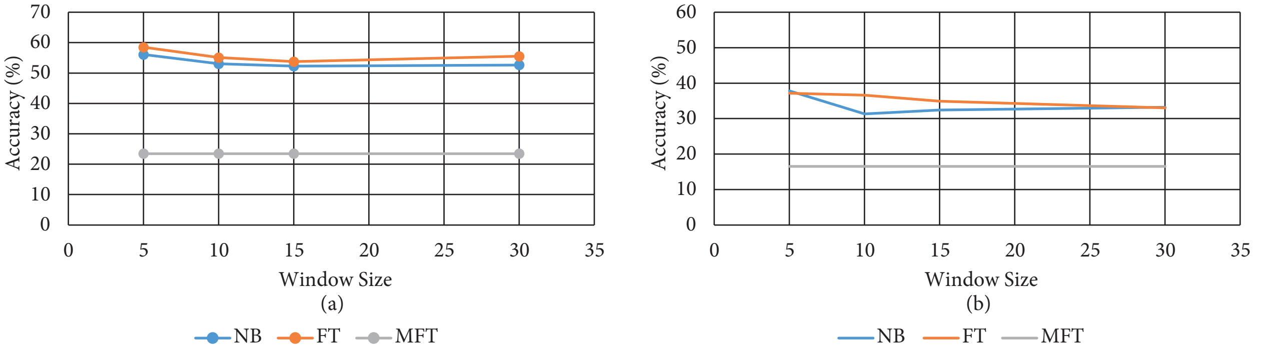


Figure 6. (a) Naïve Bayes (NB) and functional tree (FT) averages of Turkish nouns. (b) NB and FT averages of Turkish verbs. (MFB: most frequent sense baseline).

5.2. Results of the collocational features

The summarized properties in Section 4 have been used as collocational features. These are POS information, root of the headword, and neighbor words for the selected window size (i.e. words within ± 4 positions of the target word). The feature set contains all major POS types (nouns, verbs, adjectives, etc.), minor subparts, and root words in the specified neighborhood range. The total vector size has been calculated as 1081 as explained in Section 4. Table 2 presents the average accuracy results of the noun and verb sets for local collocational features. Naïve Bayes, instance based learning, decision tree, functional tree, and support vector machines (NB, IBk, J48, FT, SVM) have been used for the experiments. MFB represents the most frequent sense baseline.

Table 2. Average accuracy (%) results for collocational features.

Type	MFB	Algorithm				
		NB	IBk	J48	FT	SVM
Noun	33.47	60.59	53.87	61.04	73.47	68.95
Verb	23.60	46.51	43.12	65.92	67.26	58.55

Additional experiments have been conducted to determine the effect of diverse morphological features on the results. We took three inflectional marker groups that are widely used in Turkish as suffixes to express: (1) number/person agreement: +A1sg, +A2sg, +A3sg, +A1pl, +A2pl, +A3pl. (2) possessive agreement: +P1sg, +P2sg, +P3sg, +P1pl, +P2pl, +P3pl. (3) case: +Nom, +Acc, +Dat, +Abl, +Loc, +Gen, +Ins. There are six markers for the first two groups. The “sg” and “pl” stand for singular and plural. Ax is used to express person type (i.e. A1sg: I, A2sg: you, A3sg: he/she, A1pl: we, A2pl: you, A3pl: they). Possessive suffixes are used in the same manner. The words may get seven inflectional case markers (nominative, accusative, dative, ablative, locative, genitive, instrumental). We use the below feature sets (FSs) to get the accuracy results of noun and verb groups. For the experiments except “all features”, each morphological feature group is removed to investigate the degradation in results.

1. All features - All
2. All features except number/person agreement – FS1
3. All features except possessive agreement – FS2
4. All features except case – FS3
5. All features except word root – FS4

The experiments have been repeated for four different feature sets to determine the most effective one on removing sense ambiguity. Figures 7(a) and 7(b) display the accuracy results for the given sets. The results show that word roots and the case marker are the effective groups to disambiguate senses.

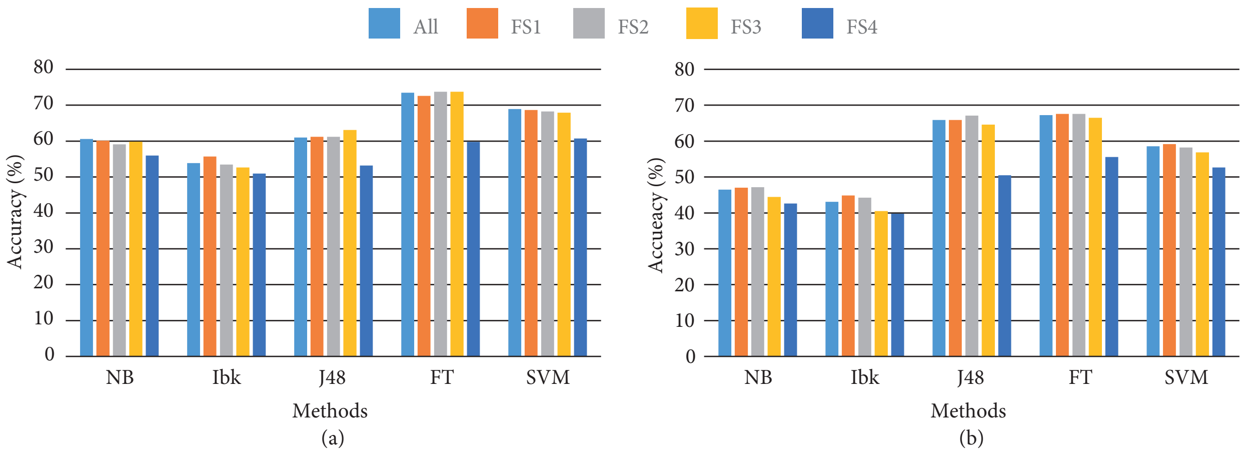


Figure 7. (a) Investigation of feature groups on Turkish noun set. (b) Investigation of feature groups on Turkish verb set. (FS: feature set).

5.3. Results of the bag-of-words features

The steps in Section 4 have been followed to apply BoW features. After removing surface forms by using morphological analysis and disambiguation tools, we determined the features. These features have been obtained by selecting the most frequent content words as mentioned in the previous section. We selected the most frequent 100 and 75 content words as features for the noun and verb sets, respectively. The window size has been taken as ± 5. In other words, the vectors have been encoded considering the number of features in this range. We used 10-fold cross validation to evaluate the average accuracy of the ML algorithms. The accuracy results of the BoW features are given in Table 3.

Table 3. Average accuracy (%) results for bag-of-words features.

Type	MFB	Algorithm				
		NB	IBk	J48	FT	SVM
Noun	33.47	58.09	46.32	59.42	58.89	57.79
Verb	23.60	44.79	37.18	46.69	44.73	43.58

5.4. Joint setting of feature sets

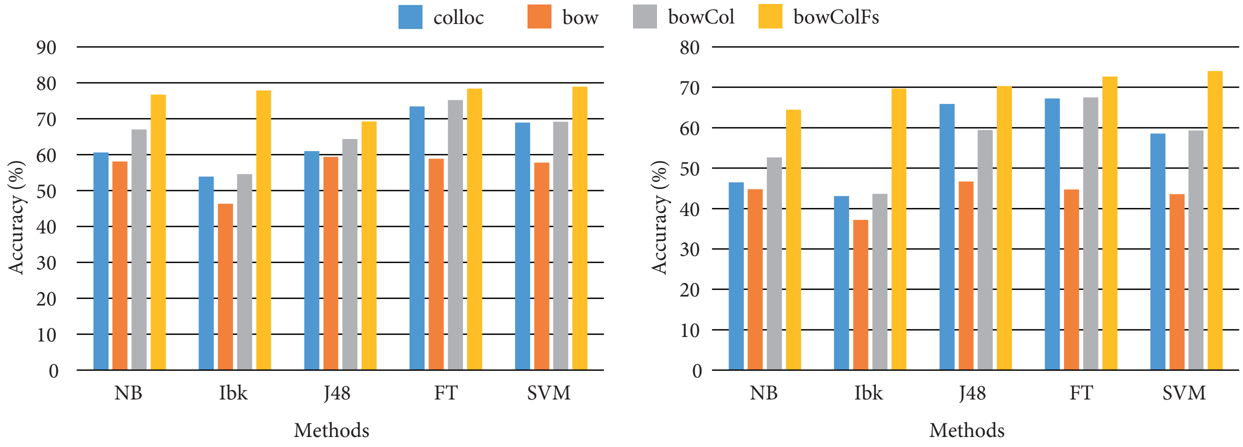
In the scope of this study, collocational and BoW features have been investigated on the TLSD using two settings: (1) separate usage of feature groups, (2) joint features combined using collocational and BoW features. The results show that the collocational features yield better results than do the BoW features. Table 4 summarizes the accuracy results of collocational features, BoW features, and joint setting of collocational and BoW features for the noun group in respective rows. The last row shows the accuracy results of collocational and BoW features after applying feature selection. Figure 8(a) shows the results of the noun set in graphical view. The “colloc” and “bow” features stand for collocational and bag-of-words features respectively. Table 5 and Figure 8(b) summarize the similar accuracy results for the verb set.

Table 4. Comparison results of average accuracy (%) for Turkish nouns.

	NB	IBk	J48	FT	SVM
ColF	60.59	53.87	61.04	73.47	68.95
bowF	58.09	46.32	59.42	58.89	57.79
ColF + bowF	67.05	54.62	64.37	75.18	69.14
ColF + bowF + FS	76.71	77.89	69.22	78.42	78.91

Table 5. Comparison results of average accuracy (%) for Turkish verbs.

	NB	IBk	J48	FT	SVM
ColF	46.51	43.12	65.92	67.26	58.55
bowF	44.79	37.18	46.69	44.73	43.58
ColF + bowF	52.68	43.63	59.42	67.53	59.29
ColF + bowF + FS	64.46	69.71	70.29	72.70	74.03



(a) (b)

Figure 8. (a) Accuracy (%) of feature sets for Turkish nouns. (b) Accuracy (%) of feature sets for Turkish verbs.

The results show that joint setting of collocational and BoW features yields the best results with feature selection. When separate features are chosen, the results of collocational features are better than the results of BoW features.

5.5. Comparison with previous work

The feature sets that we investigated on the TLSD have also been tested on a different corpus that was used in previous WSD efforts [32] on Turkish. We repeated our experiments on the METU-Sabancı Turkish Treebank corpus, which includes Turkish noun and verb groups. Although some of the ambiguous words were common, most of them were different from the ones in TLSD. We used completely the same portion (highly ambiguous 10 Turkish nouns and 10 verbs) of the METU-Sabancı Turkish Treebank, which was previously used in [32]. It is also a part of the Senseval project and both the training and test portions of this dataset can be accessed online.

For the experiment in [32], polysemous Turkish nouns and verbs have been determined to employ the "Turkish Lexical Sample Task" on the METU-Sabancı Turkish Treebank corpus. For this task there are approximately 100 examples tagged per word. It is also reported that some ambiguous words have fewer examples. As a solution they were eliminated or similar examples were added from external resources.

In [32], some of the features that were employed for previous and subsequent words are word root, POS information, inflected POS, tags for ontology level, case marker, possessor, and relation. For the target word, only the root word, case marker, possessor, relation, and the POS information have been taken into consideration. Fine and coarse grained (FG and CG) sense numbers have been used in their experiments. The precision and recall values of [32] for the Turkish noun class have been given as 0.15 and 0.50, respectively. The precision and recall results of the verb class have been reported as 0.10 and 0.38. Our results for the same Treebank corpus are shown in Table 6. The precision results that we obtained are 0.46 and 0.54 points above the previous work for the noun and verb groups, respectively. Similarly, our recall values are 0.07 and 0.28 points above the considered study.

Table 6. Average precision-recall values for FG senses.

	Precision	Recall
Noun	0.61	0.57
Verb	0.64	0.66
Average	0.63	0.62

Our feature selection strategy is better than that of the previous work. We used a larger window size than the considered work did [32]. They used stem, possessor, case marker, ontology level, part of speech, file id, and sentence number as features. We avoided using features such as “file id” and “sentence number”. An important difference between this work and [32] is that we have not used syntactic features in any part of the study. On the other hand, the considered work utilizes these features by using relations between the target and previous/subsequent words. Ontology level is the other kind of information in [32] that we have not utilized.

It has been noted that the average of the verb group is higher than that of the noun set. It is thought that this results from the limited size of the training portions of the dataset. On the other hand, the average training size of the verb set is a bit larger. Another point that must be noted is the length of the samples. The size of samples in Treebank may be too small. As a result, it has been inevitable to omit some part of the window information for some cases.

6. Conclusion

It is known that the fundamental features extracted from context words can accurately isolate a given sense, and there are many features that can contribute to the meaning of a given word. In the scope of this work we conducted several experiments to determine the impact of feature set selection and the windowing scheme. We found the accuracy values for collocational and BoW features. Then we utilized both feature sets together. According to the results, collocational features are more effective than BoW features in the disambiguation of word senses. However, the averages of the joint setting of two feature sets are better than that of the collocational features. Our additional experiments on BoW features show that different number of content words may be sufficient as features to encode noun and verbs sets. We investigated the optimal window size by using BoW features. The results suggest that the optimal window size on average is 5 for the noun and verb sets.

Considering the agglutinative property of the Turkish language, we investigated the effect of diverse morphological feature sets such as number/person agreement, possessive agreement, and the case marker. The accuracy results have been observed by eliminating each feature set consecutively. Our experiments on the

METU-Sabancı Turkish Treebank yielded better precision and recall values than previous work did. Considering the results of this work is important since the experiments have been carried on the same corpus and the ambiguous set of Turkish words. The overall results indicate that selecting significant features is crucial for agglutinative languages such as Turkish. The appropriate selection of features can contribute more than the contribution of using different learning algorithms.

References

[1] Bhala RV, Abirami S. Trends in word sense disambiguation. Artif Intell Rev 2014; 42: 159-171.

[2] Jurafsky D, Martin JH. Speech & Language Processing. 2nd ed. Pearson Education India, 2000.

[3] Zhou X, Han H. Survey of Word Sense Disambiguation Approaches. In: Proceedings of the 18th International FLAIRS Conference; 15–17 May 2005; Florida, USA. pp. 307-313.

[4] Agirre E, Lacalle OL, Martinez D. Exploring feature spaces with svd and unlabeled data for Word Sense Disambiguation. In: Proceedings of the Conference on Recent Advances on Natural Language Processing; 21–23 September 2005; Borovets, Bulgaria.

[5] Cancho RF. The meaning-frequency law in Zipfian optimization models of communication. In: arXiv preprint arXiv:1409.7275, 2014.

[6] Ide N, Véronis J. Introduction to the special issue on word sense disambiguation: the state of the art. Comput Linguist 1998; 24: 2-40.

[7] Orhan Z, Altan Z. Determining effective features for word sense disambiguation in Turkish. IU-JEEE 2011; 5: 1341-1352.

[8] Ilgen B, Adali E, Tantug AC. The impact of collocational features in Turkish Word Sense Disambiguation. In: IEEE 16th International Conference on Intelligent Engineering Systems; 13–15 June 2012; Lisbon, Portugal.

[9] Dang HT, Palmer M. Combining contextual features for word sense disambiguation. In: Proceedings of the ACL-02 workshop on Word sense disambiguation: recent successes and future directions; 2002; Philadelphia. Association for Computational Linguistics, 2002. pp. 88-94.

[10] Dang HT, Chia C, Palmer M, Chiou F. Simple features for Chinese word sense disambiguation. In: Proceedings of Coling-02 19th International Conference on Computational Linguistics; 2002; Taipei, Taiwan.

[11] Agirre E, Lacalle OL, Martínez D. Exploring feature set combinations for WSD. In: Proceedings of the SEPLN, 2006.

[12] Leech G. 100 million words of English: the British National Corpus (BNC). Language Research 1992; 28: 1-13.

[13] Orhan Z, Altan Z. Effective features for disambiguation of Turkish verbs. In: International Enformatika Conference IEC'05; 26–28 August 2005; Prague, Czech Republic. Watermark, 483. pp. 182-186.

[14] Suárez A, Palomar M. Feature selection analysis for maximum entropy-based wsd. In: Computational Linguistics and Intelligent Text Processing; 2002. Springer. pp. 146-155.

[15] Ng HT, Lee HB. 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.

[16] Miller GA, Chodorow M, Landes S, Leacock C, Thomas RG. Using a semantic concordance for sense identification. In: Proceedings of the Workshop on Human Language Technology. Association for Computational Linguistics, 1994. pp. 240-243.

[17] Leacock C, Towell G, Voorhees E. Corpus-based statistical sense resolution. In: Proceedings of the Workshop on Human Language Technology. Association for Computational Linguistics, 1994. pp. 260-265.

[18] Bruce R, Wiebe J. Word-sense disambiguation using decomposable models. In: Proceedings of the 32nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics, 1994. pp. 139-146.

[19] Specia L, Srinivasan A, Joshi S, Ramakrishnan G, Nunes MDGV. An investigation into feature construction to assist word sense disambiguation. Mach Learn 2009; 76: 109-136.

[20] Montoyo A, Suárez A, Rigau G, Palomar M. Combining knowledge-and corpus-based word-sense-disambiguation methods. J Artif Intell Res 2005; 23: 299-330.

[21] Scott S, Matwin S. Feature engineering for text classification. In: ICML; 1999. pp. 379-388.

[22] Chodorow M, Leacock C, Miller GA. A topical/local classifier for word sense identification. Comput Humanities 2000; 34: 115-120.

[23] Navigli R. Word sense disambiguation: a survey. ACM Comput Surv 2009; 41: 1-69.

[24] Altintas E, Karsligil E, Coskun V. The effect of windowing in word sense disambiguation. In: Computer and Information Sciences-ISCIS 2005; 2005. Springer Berlin Heidelberg. pp. 626-635.

[25] Banerjee S, Pedersen T. An adapted Lesk algorithm for word sense disambiguation using WordNet. In: Computational linguistics and intelligent text processing. Springer, 2002. pp. 136-145.

[26] Fellbaum C. WordNet: An Electronic Lexical Database. Cambridge, MA, USA: MIT Press, 1998.

[27] Yarowsky D. One sense per collocation. In: Proceedings of the Workshop on Human Language Technology. Association for Computational Linguistics, 1993. pp. 266-271.

[28] Ilgen B, Adali E, Tantug AC. Building up lexical sample dataset for Turkish Word Sense Disambiguation. In: IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA); July 2012; Trabzon, Turkey.

[29] Turkish Language Association. Güncel Türkçe Sözlük. Ankara, Turkey: TDK Publishing, 2005.

[30] Oflazer K, Say B, Hakkani-Tür DZ, Tür G. Building a Turkish treebank. In: Anne Abeillé, editor. Treebanks. Amsterdam, Netherlands: Kluwer Academic Publishers, 2003. pp. 261-277.

[31] Atalay NB, Oflazer K, Say B. The annotation process in the Turkish treebank. In: Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC); 2003.

[32] Orhan Z, Çelik E, Demirgüç N. SemEval-2007 task 12: Turkish lexical sample task. In: Proceedings of the 4th International Workshop on Semantic Evaluations; 2007. Association for Computational Linguistics. pp. 59-63.

[33] Oflazer K. Two-level description of Turkish morphology. Literary and Linguistic Computing 1994; 9: 137-148.

[34] Yuret D, Türe F. Learning morphological disambiguation rules for Turkish. In: Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics; 2006. pp. 328-334.

[35] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter 2009; 11: 10-18.

[36] İlgen B, AdalıE, Tantuğ AC. A comparative study to determine the effective window size of Turkish Word Sense Disambiguation systems. In: Information Sciences and Systems 2013; 28–29 October 2013. Springer. pp. 169-176.