Constructing a WordNet for Turkish Using Manual and Automatic Annotation

RAZIEH EHSANI, ERCAN SOLAK, and OLCAY TANER YILDIZ, Işık University

In this article, we summarize the methodology and the results of our 2-year-long efforts to construct a comprehensive WordNet for Turkish. In our approach, we mine a dictionary for synonym candidate pairs and manually mark the senses in which the candidates are synonymous. We marked every pair twice by different human annotators. We derive the synsets by finding the connected components of the graph whose edges are synonym senses. We also mined Turkish Wikipedia for hypernym relations among the senses. We analyzed the resulting WordNet to highlight the difficulties brought about by the dictionary construction methods of lexicographers. After splitting the unusually large synsets, we used random walk–based clustering that resulted in a Zipfian distribution of synset sizes. We compared our results to BalkaNet and automatic thesaurus construction methods using variation of information metric. Our Turkish WordNet is available online.

CCS Concepts: • **Computing methodologies → Lexical semantics**;

Additional Key Words and Phrases: WordNet, synset, Turkish

ACM Reference format:
Razieh Ehsani, Ercan Solak, and Olcay Taner Yildiz. 2018. Constructing a WordNet for Turkish Using Manual and Automatic Annotation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 17, 3, Article 24 (April 2018), 15 pages.
https://doi.org/10.1145/3185664

1 INTRODUCTION

A word may have more than one sense. Additionally, a sense may be shared among different words. In natural language processing (NLP), finding the words that share a sense and identifying in which of their senses they mean the same thing is the task of WordNet construction.

A WordNet is a graph data structure where the nodes are word senses with their associated lemmas (and collocations in the case of multiword expressions (MWEs)) and edges are semantic relations between the sense pairs. Usually, the multiple senses corresponding to a single lemma are enumerated and are referenced as such. For example, the triple

$$\left(w_2^5, w_3^7, r_1\right)$$

represents an edge in the WordNet graph and corresponds to a semantic relation r_1 between the second sense of the lemma w^5 and the third sense of the lemma w^7. The direction of the relation is usually implicit in the ordering of the elements of the triple. For synonymy, the direction is symmetric. For hypernymy, as a convention, the first sense is an hyponym of the second.

Authors' addresses: R. Ehsani, E. Solak, and O. T. Yildiz, Department of Computer Engineering, Işık University, Meşrutiyet köyü, Üniversite sokak, no: 2, Şile, İstanbul, 34980 Turkey; emails: {razieh.ehsani, ercan.solak, olcay.yildiz}@isikun.edu.tr.

https://doi.org/10.1145/3185664

The most pervasive relation in a WordNet is the synonymy. We take two lemmas as synonyms if there is a linguistic context in which they are interchangeable [13].

WordNets provide semantic ontologies that are used as inputs to many automated document analysis tasks, such as summarization and classification [8, 9, 21].

Constructing a WordNet is a labor-intensive undertaking. Annotating in a WordNet requires lexicographical competency and familiarity with the modes of use of words in different domains.

The first WordNet project was Princeton WordNet (PWN), which was initiated in 1995 by Miller [12]. Over time, PWN evolved to become a comprehensive relational representation of the word senses of English. Currently, the latest release of PWN, version 3.1, has 117,000 synsets and 206,941 word-sense pairs. A more detailed history and description of PWN is given in Fellbaum [7].

Shortly after the release of PWN, WordNets for other languages were constructed. Many WordNets for other languages use the leverage PWN by translating its synsets and extending where necessary. For example, Finnish WordNet has the same number of synsets as PWN [10]. Version 3.0 of Polish WordNet, plWordNet, is larger than PWN by about 1,000 more words [15].

EuroWordNet (EWN) [20] is a multilingual WordNet developed for seven European languages. Arabic WordNet [4] follows a similar approach to EWN and focuses on manually extracting sets of concepts and maximizing compatible relations between Arabic WordNet and other WordNets. For Balkan languages, BalkaNet [19] is the most comprehensive work to date. The current state of the development of various WordNets can be found of the Web site for the Global WordNets Association [2].

For the Turkish WordNet part of BalkaNet [3], the researchers automatically extracted synonyms, antonyms, and hyponyms from a monolingual Turkish dictionary. A similar monolingual dictionary mining approach was recently used to extract hypernyms for Russian WordNet [1].

In BalkaNet, developers started with a core set of words that are deemed to be common across several languages. We followed a bottom-up approach in our development. Rather than starting from a core set of lemmas and the relations among them, we started with the whole set of lemmas in a monolingual dictionary of Turkish. This is particularly in contrast to the approach used in BalkaNet and several similar WordNets. Starting from a common set for multiple languages as was done in BalkaNet poses problems for translating lemmas in one language to other. For example, a sense that is expressed using a single word form in one language can only be expressed using a nonidiomatic phrase in another language. For example, the word "payday" in PWN is translated as the phrase "maaş ödeme günü" (literally, salary paying day) in Turkish BalkaNet. This phrase is not a collocation to warrant its inclusion in a monolingual Turkish dictionary. To the best of our knowledge, no other WordNet construction efforts have used such a bottom-up approach as ours.

In the work detailed in the present article, we kept the data format compatible with that used in BalkaNet. Therefore, our final data can easily be used in extending BalkaNet to cover a larger number of synsets. We named our WordNet *KeNet* using the two initial letters of "kelime" (word in Turkish).

In our approach to WordNet construction for Turkish, we mined a comprehensive online dictionary of Turkish for synonym candidates. We then manually annotated the whole set of candidates to verify the synonymy and pair the particular senses of the verified synonyms. Thus, we obtained a graph where the nodes are senses and the edges are synonymy relations. We found the clusters in this graph and arrived at synsets. We compared the resulting set of synsets to an automatic thesaurus that we constructed and the smaller set of synsets obtained in BalkaNet.

Compared to BalkaNet, our approach yields a WordNet that is larger and more consistent. Moreover, as we detail in the rest of the article, our double manual annotation increases the reliability of the resulting synsets. We also mined Turkish Wikipedia for hypernym relations that increased the set of such relations obtained using only a dictionary.

The rest of the article is organized as follows. In Sections 2 and 3, we describe the lexical resources that we use in our work. In Sections 4, 5, and 6, we give the details of the manual annotation and synset construction. Section 7 concludes with discussions of the results and suggestions for further research.

2 LEXICAL RESOURCE

The main lexical source for KeNet is the Contemporary Dictionary of Turkish (CDT) (Güncel Türkçe Sözlük) published online and in paper by the Turkish Language Institute (TLI) (Türk Dil Kurumu), a government organization. Among other literary and academic works, the TLI publishes specialized and comprehensive dictionaries. These dictionaries are often taken as an authoritative reference by other dictionaries. The online version of the CDT contains 65,944 lemmas. Although the TLI publishes a separate dictionary of idioms and proverbs, the CDT still contains some MWE entries that have idiomatic senses. In Section 3.2, we discuss how we handle MWEs in KeNet.

The first edition of the CDT was published in 1945. Since then, it has been revised and updated many times. Currently, the CDT's 2011 print edition is in circulation. Its online version is revised more often.

In our work, we used a reduced snapshot of the CDT online.

The CDT has a pretty straightforward structure without any special markups. For example, synonyms are not marked up but are instead embedded within the sense definitions. Homonyms of a word are enumerated under the same entry for the lemma. Most lemmas have no homonyms. The entries with the largest number of homonyms have five. These are "bel" (sign, waist, sperm, spade, sound magnitude unit) and "bar" (a folk dance, pub, pressure unit, bitterness in mouth, stick).

The fields of an entry in the CDT are as follows (multiplicity and optionality of the field values are given at the end of each field description):

Alternation: Indications of orthographic changes in suffixation. Optional.

Domain: Whenever an entry has a technical sense, its domain is given. Multiple, optional.

Default POS: Most common POS of the lemma. It can be empty for MWEs. It has one of the following values: verb, auxiliary verb, conjunction, postposition, common noun, adjective, pronoun, adverb, proper noun, exclamation. Multiple, optional.

Origin: Source language for loan words. It can be multiple valued for MWEs. Multiple, optional.

Pronunciation: It is often given in cases of loan words where the stress cannot be regularly predicted. Optional.

Context: Indication of usages such as argot and mockery. Multiple, optional.

Senses: An enumerated list of senses. Each sense might have its own POS and domain fields when they are different than those of the default.

Even though the CDT is the main authoritative lexical resource in Turkish, it poses some difficulties when used for NLP tasks. In the following, we examine some of the issues that we encountered in out WordNet construction.

2.1 Sense Granularity

The CDT is prone to some of the common lexicographic problems that afflict many dictionaries. For WordNet construction, the most relevant is the proliferation of senses where the distinction among the senses are debatable.

For example, the senses of the word "yüz" (hundred) are given in the CDT as follows:

(1) The name of the number after 99.
(2) The name of the numerals 100 and C that denote this number.

(3) Ten times 10, one more than 99.
(4) A word that when used together with "times" and "fold" exaggeratedly expresses the multitude of something done.

This particular entry is somewhat extreme, but it highlights the problems of lexicography in the CDT. The first three senses can easily be collapsed without any loss of precision into a single sense with a short definition "the number 100." Only the fourth sense is sufficiently distinct and similar to its use in English.

In contrast, the online Oxford dictionary [16] lists only a single sense with the definition "the number equivalent to the product of ten and ten; ten more than ninety; 100," thus practically collapsing the first three senses given in the CDT. The finer distinctions among the senses of "hundred" are given as usages under its single main sense in the online Oxford dictionary.

The sense granularity of PWN and its effect on sense clustering tasks was investigated in Palmer et al. [14] and Snow et al. [18].

2.2 Productive Derivations

Turkish is an agglutinative language with a highly productive derivational morphology.

The derivations pose interesting problems for lexicographers. An important problem is to decide whether to include a derivation as a separate entry in the dictionary. An intuitive decision boundary is to distinguish the cases where the derived form undergoes a sense drift away from the one that the derivational morpheme nominally entails. If the drift is so large that the sense of the derivation cannot be inferred from those of the root and the suffixes, then a new sense needs to be added to the dictionary.

There are about 40 highly productive derivational suffixes in Turkish. One particular example is the deverbal noun suffix -mA, with the semantics "the action of the verb." The CDT has about 5,400 entries for deverbal nouns with suffix -mA where the definition has the single obvious sense of "the act of verb." Similarly, the CDT includes separate entries for causative and reflexive forms of verbs. For example, the CDT has an entry for the verb "sor" (ask), as well as separate entries for the deverbal noun "sor-ma" (the act of asking), "sor-dur" (cause to ask), and "sor-ul" (be asked). Each of those entries has the single obvious sense that can be trivially inferred from the semantics of the root "sor" and the suffixes -mA (deverbal noun), -DIr (causative), and -Il (passive).

In parsing the dictionary, we had to decide whether to include these obvious productions with single senses as nodes in KeNet or leave them out to be dealt with derivational morphology. In the initial version of KeNet, we decided to keep these derived nodes as singleton synsets.

3 PROCESSING THE DICTIONARY

In this section, we give the details of preparation task that we performed before we presented the human annotators with synonym candidates.

3.1 Synonym Candidates

CDT definitions include synonyms in many cases. Although the synonyms are not specially marked, the structures of most definitions are consistent enough to enable the development of heuristics for automated synonym extraction. Synonyms are usually listed toward the end of a definition, separated by commas. In many cases, the definition itself is a single word or a MWE, yielding unambiguously a synonym candidate. In other cases, we slice the definitions at commas. We eliminate the slices that do not have entries of their own in the dictionary. What remains is a list of synonym candidate lemmas associated with a dictionary entry. We store these candidate pairs for manual annotation as detailed in Section 3.3.

Table 1. Auxiliary Verbs in Turkish and Their Frequencies

Verb Stem	Closest Translation	MWE Count in the CDT
et	do	1,227
ol	be	298
ver	give	88
gel	come	85
kal	stay	58
git	go	51
yap	do	45
geç	pass	43
getir	bring	30
göster	show	20
dur	stay, stand	11
kıl	render	5
yaz	write	2
eyle	do	1
Total		1,964

3.2 Handling MWEs

Many dictionaries contain MWEs that are idiomatic to some degree. The CDT is no exception. In pruning the dictionary, we had to decide which of these MWEs to keep and which ones to discard, possibly relegating them to a specialized graph of idioms and their usages.

For Turkish, of particular interest are the verbal compounds. Verbal compounds are formed by the combination of a (possibly case-marked) noun or adjective with one of the auxiliary verbs. Most common auxiliary verbs in Turkish verbal compounds are “etmek” (to do) and “olmak” (to be). The CDT lists 14 verbs as having at least one sense in which it has an auxiliary function. Auxiliary verbs in Turkish are listed in Table 1.

Moreover, verb compound formation is the main mechanism through which foreign verbs are borrowed into the language. Basically, the infinitive form of the borrowed word in the foreign lexicon is compounded with “etmek” to construct the MWE infinitive form. Examples are “tasnif etmek” (from Arabic, “tsnyf,” to classify), “lanse etmek” (from French, “lancer,” to launch), and “dizayn etmek” (from English, to design).

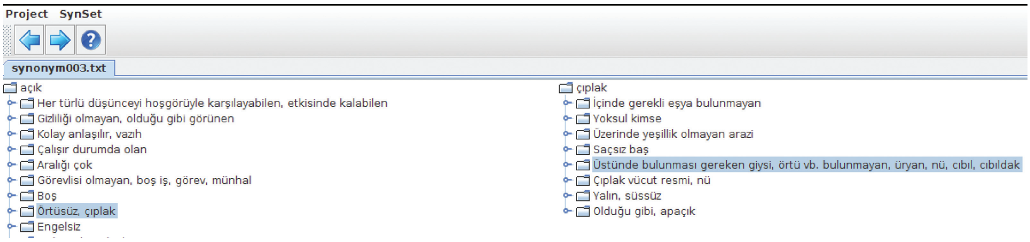
Such MWEs appear as lexical entries in the CDT. They also commonly appear in sense definitions. In mining for synonym candidates in CDT definitions, we included an MWE as a synonym candidate only if it has a dedicated entry and one or more senses listed.

The CDT also includes MWE templates as separate entries. An example is the entry “duygusu-uyandırmak” (literally, “feeling of to wake,” meaning, “to arouse a feeling of”). There are about 50 of such template entries. We discarded these in our construction, as it not possible the encounter them in template form in any sense definition.

3.3 Manual Annotation

In processing the CDT, we sliced the sense definitions at the commas. Thus, we expect to find synonym literals among the slices. Of course, not every slice is a synonym. We used the following procedure to manually select the synonym sense when present.

Let $C(l)$ denote the set of such slices extracted from the sense definitions of lemma l. Let $S(l)$ denote the set of sense definitions of lemma l. For each l_i in the CDT and for each j such that



açık
- Her türlü düşünceyi hoşgörüyle karşılayabilen, etkisinde kalabilen
- Gizliliği olmayan, olduğu gibi görünen
- Kolay anlaşılır, vazıh
- Çalışır durumda olan
- Aralığı çok
- Görevlisi olmayan, boş iş, görev, münhal
- Boş
- Örtüsüz, çıplak
- Engelsiz

çıplak
- İçinde gerekli eşya bulunmayan
- Yoksul kimse
- Üzerinde yeşillik olmayan arazi
- Saçsız baş
- Üstünde bulunması gereken giysi, örtü vb. bulunmayan, üryan, nü, cıbıl, cıbıldak
- Çıplak vücut resmi, nü
- Yalın, süssüz
- Olduğu gibi, apaçık

Fig. 1. Screenshot of the synset reduction tool.

$l_j \in C(l_i)$, we present the human annotator the sets $C(l_i)$ and $C(l_j)$ as two lists. The annotator picks one sense from each, thus creating a synonym sense pair. The annotator may choose not to pair any of the senses. This means that the annotator judges that the lemmas l_i and l_j are not synonymous in any of their senses.

Figure 1 shows a screenshot of the synonym pairing tool that we constructed for the manual annotation task. It shows the sets C(açık) and C(çıplak) for lemma "açık" and "çıplak." In this particular case, the annotator paired the eighth sense of "açık" with the fifth sense of "çıplak."

Note that the eighth sense of "açık" has "çıplak" as a comma separated slice, and that is the reason these two sets are shown the annotator. We instructed the annotators to disregard this bit as a pairing clue and to use their own linguistic competence in deciding which senses to pair or whether to pair at all.

We annotated each pair of synonym candidate lemmas twice using two different annotators. There were a total of nine annotators. Each annotator was given a different segment of the dictionary. The annotators were native Turkish speakers in their senior university years.

Then we determined the set of pairs where two annotators disagreed. An expert annotator went over this disagreement set and reannotated the pairs, not necessarily agreeing with one of the annotators. In the case of agreements, the expert did not modify the pair. The expert annotator is the first author of the present article, a native speaker of Turkish.

The total number of annotated pairs is 49,774. Of these, 42,615 had annotator agreements. In 92.15% of the pairs with disagreements, the expert annotator agreed with one of the annotators. For the rest, the expert chose a different pair that we accepted as the authoritative choice.

The annotation tool that we used in available for download on the KeNet Web page [6].

4 SYNSET CONSTRUCTION

After manually pairing lemmas with their matching senses, we collect the pairs to form maximal synsets. First, we decided on a procedure on how to treat the pairs where annotators disagreed about the sense match.

4.1 Interannotator Agreement

The simplest synset construction method is to find the connected components of the graph where the nodes are senses and the edges are the pairs of senses marked by the annotators as matching. Such a simple construction assumes that all edges have the same confidence levels. Actually, the edges differ in their confidence strength. If the two teams agree on a pair, the confidence level is high. On the other end of the scale, if they disagree and the expert annotator chooses a pairing different than both, the confidence level is lowest. In between, the expert annotator might concur with one of the teams.

Table 2. Interannotator Agreement Statistics

	A & B	A & E	B & E	E Only	Total
Pairs (#)	42,615	1,759	4,838	562	49,774
Agreement Percentage	85.62	3.53	9.72	1.13	100

Table 2 gives the statistics of the interannotator agreement statistics for the pairs. A and B denote the annotators, and E denotes the expert.

In constructing the synsets, we took an edge to be valid if either both annotators marked the edge or the expert marked the edge. The first condition corresponds to the first column in Table 2. The second condition corresponds to its next three columns.

To measure the interannotator agreement against agreement by change, consider two lemmas l_i and l_j presented to two annotators for pairing. Let S_i and S_j be the sets of senses of the lemmas l_i and l_j, respectively. The probability of chance agreement for this pair is given as $p_c(i,j) = 1/(|S_i||S_j| + 1)$. Averaged over all pairs, we obtain the probability of chance agreement as $p_c = 0.28$. Reading off the agreement probability from the first column of Table 2 as $p_a = 0.85$, we calculate the kappa measure as

$$\kappa = \frac{p_a - p_c}{1 - p_c} = 0.79.$$

We illustrate a common source of disagreements with an example. For the lemma "akbaba," the CDT gives three senses. The definition for the first sense is the description of the animal "vulture." The second sense definition is a single word, "ihtiyar" (elderly, old person). The last sense definition is the phrase "çıkarıiçin başkalarını sömüren" (someone who exploits others for his or her own benefit). The annotators are asked to pick a pair of senses from three senses of "akbaba" and five senses of "ihtiyar." Among the sense definitions of "ihtiyar," two are quite close: "old person" and "elderly." Whereas one annotator chose the first, the other annotator chose the second sense, creating a disagreement. Thus, the close senses of a lemma is a common source of disagreement.

4.2 Synset Statistics

Once we have the pairs of senses matched by the annotators, we constructed the synsets by finding the connected components of the undirected graph where the nodes are the senses and the edges are the manual sense pairings. In Figure 2, we give the distribution of the sizes of synsets.

Considering the logarithmic scale in Figure 2, the synset sizes displays a Zipfian distribution, which is typical of some linguistic observations [22]. Note that most of the senses are singletons. There are 49,361 synsets with only a single sense. Additionally, there is a huge synset with 7,906 senses. Obviously, this cannot represent the true state of the sense relations in Turkish.

There are a few reasons for obtaining such a big synset. The main one is the sense drift introduced in the definitions of the CDT. In the CDT, often a definition for a sense cites lemmas with close senses. In some cases, this is done to confine and illustrate the sense by providing several close senses and emphasizing their intersections. However, taken in isolation, each such sense represents a small drift away from the original sense. When the annotators are presented such explanatory senses as synonym candidates, they tend to mark them as synonyms. When done in tandem, this drift creates the huge artificial synset that we observed earlier.

There are other big synsets due to the sense drift. The second biggest synset has 140 lemmas. The huge difference between the sizes of the largest and the second largest synsets indicates the presence of a property that joins smaller synsets. In terms of the graph structure, there are edges that connect small, densely connected subgraphs. To explore this issue further, we used random

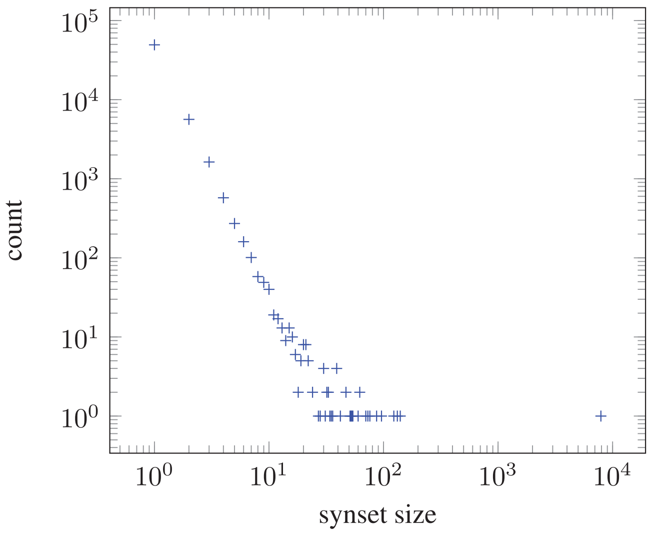


Fig. 2. Distribution of synset sizes.

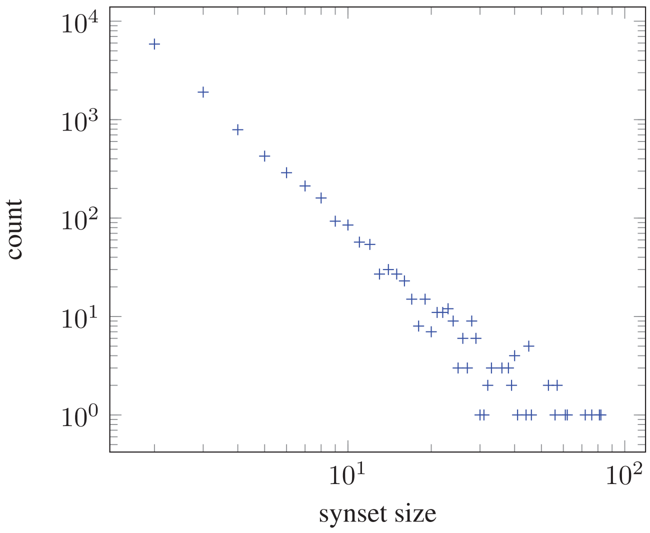


Fig. 3. Distribution of synset sizes after recursive partitioning with random walk.

walk–based partitioning of the synonym candidate graph [17]. We recursively reclustered the synsets that are larger than 100 until all synsets are smaller than 100. The final distribution of synset sizes are given in Figure 3. The figure displays the sizes only for synsets that are not singletons.

4.3 Semantic Relations

In finding relations among the synsets in KeNet, we confined the set of relations to the following three: antonym, hypernym, and hyponym. We determined the candidate pairs for these relations by rule-based processing of the CDT and Turkish Wikipedia. Human annotators reviewed the

Table 3. Statistics for the Semantic Relations

Relation	Source	Count
Antonym	CDT	376
Hypernym	CDT	1,420
Hypernym	Wikipedia	2764

Table 4. Example Patterns for Hypernym Candidates

Pattern in Turkish	Pattern in English
SUP-A verilen genel (ad,isim)DIr	is the general name given to SUP
bir SUP-DIr	is a SUP
SUP kavramlarından birisidir	is one of SUP concepts
SUP (çeşidi, türü, birisi)DIr	is a (kind, one of) SUP
SUBlArin (bütünü, tümü)dür	is the whole of SUBs

automatic results and eliminated the false and ambiguous candidates. Table 3 summarizes the results.

Since hypernym and hyponym relations are inverses of each other, we considered them under the same set. Thus, if we detect a pair (a, b) as a being a hypernym of b, we take that we also detected b as a hyponym of a.

The rule-based search generates a list of candidate pairs between the lemmas, not senses. When reviewing the candidates, the human annotators determined the pair of particular senses for which the relation holds.

For the rest of this section, we give the details for each type of rule-based search.

4.4 Antonyms

We use the characteristics of antonyms in searching for antonym candidates within the CDT. We search for patterns that represent opposition in sense definitions. The most common opposite pattern in the CDT is "*l* karşıtı" (opposite of l), where *l* is a lemma.

4.5 Hypernyms and Hyponyms

Finding hypernym relations in descriptive text is easier than it is for hyponyms. In a sense definition in the CDT, hypernym sense is often referred to with the assumption that the more abstract sense of the hypernym would be more readily available in the mental lexicon of the reader. Definitions often describe the peculiarity of the present entry and toward the end mention the hypernym. For example, the definition of "flamingo" in the CDT is

"Leyleksigillerden, tüyleri beyaz, pembe, kanatlarının ucu kara, eti yenir bir *kuş*"
"An edible *bird* from the stork family with white and pink feathers and black wing-tips."

There are variations to the patterns of referring to the hypernyms. Some examples of patterns are listed in Table 4. Here, SUP refers to the hypernym in the pattern and SUB refers to a hyponym. Detecting hyponyms in sense definitions is more difficult, as they rarely refer to subclasses. Still, since hypernym/hyponym relations are inverses of each other, we generate hyponym relations wherever we detect hypernyms.

In addition to the CDT, we used Turkish Wikipedia in our search for hypernyms.

5 AUTOMATIC CONSTRUCTION

To compare the performance of our manual synset construction procedures, we constructed a synonym thesaurus using a fully automatic, rule-based processing of the CDT dictionary. Our aim in this comparison is to see how necessary it is to manually annotate the synonym candidate pairs extracted from the dictionary. The automatic methods that we describe in the following yield only thesaurus synsets with an extreme granularity, namely collapsing all sense definitions into a single definition when searching for synonym candidates. Therefore, we confine the comparison to the lemmas for which the CDT lists a single sense.

In this section, we give the details of automatic construction. The details of the comparison are given in Section 6.

5.1 Automatic Thesaurus

Let $S_i(w)$ denote the definition of the ith sense of the entry for lemma w as given in the CDT.

Let R denote a deterministic rule that generates the list of candidate lemmas from a given sense definition in the dictionary. An example rule would be to slice the definition at commas and keep only the slices that are cited as the dictionary entries. For illustration, consider the definition of the first sense of the lemma "boğaz" (neck) given in the CDT as "Boynun ön bölümü ve bu bölümü oluşturan organlar, imik, kursak." Under the simple comma slicing rule, we generate two synonym candidate lemma: "imik" and "kursak."

For every entry in the dictionary, we define the set $C(w)$ of candidate synonym lemma for the lemma w as

$$C(w) = \{v | \exists i, v \in R(S_i(w))\}.$$

Namely, $C(w)$ represents the candidate lemmas obtained through running rule R over all sense definitions of w.

We next define the notion of strong synonymy with respect to a dictionary D and rule R as follows.

Definition 5.1. Literals w_1 and w_2 are strongly synonymous with respect to dictionary D and rule R if

$$w_1 \in C(w_2) \wedge w_2 \in C(w_1).$$

Note that we do not distinguish among different senses of a lemma and instead consider all of the synonym candidate lemmas collected across all senses.

This definition of synonymy is very restrictive. It requires the lexicographer to be complete when they write the sense definitions and include all synonym candidates symmetrically.

A weaker definition allows longer cycles in mapping lemmas to synonym candidates. For this, we first define the longer synonym candidacy relation among lemmas. Let us define the set of n-synonym candidates $C_n(x_0)$ of the lemma x_0 as

$$\begin{aligned} C_n(x_0) = \{x_n | \exists x_1, x_2, \ldots, x_{n-1}, \text{where } x_i \neq x_j, 1 \leq i, j < n, \\ x_1 \in C(x_0), x_2 \in C(x_1), \ldots, x_n \in C(x_{n-1})\}. \end{aligned} \tag{1}$$

Combining all of the paths up to length n, we define the weakly n-synonym candidate set $\bar{C}_n$ for a lemma x_0 as

$$\bar{C}_n(x_0) = C_1(x_0) \cup C_2(x_0) \ldots \cup C_n(x_0).$$

Now we can define weakly n-synonymy.

Definition 5.2. Two lemmas w_1 and w_2 are weakly n-synonymous with respect to dictionary D and rule R if

$$w_1 \in \bar{C}_n(w_2) \wedge w_2 \in \bar{C}_n(w_1).$$

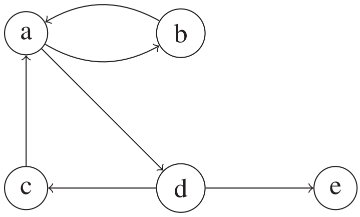


Fig. 4. Synonym candidacy relations on the word graph.

Obviously, weakly 1-synonymy is the same as strong synonymy.

It is easy to visualize these new synonymy relations if we view the lemmas as the nodes of a graph and the synonym candidacy relations as directed edges between the nodes. Figure 4 illustrates the various relations. In the graph, a and b are strong synonyms. The lemmas a and c are weakly 2-synonyms. Note that the path from c to a and the path from a to c are of different length.

Definition 5.3. Two lemmas w_1 and w_2 are weakly synonymous if there is an integer $n \geq 1$ such that w_1 are w_2 are weakly n-synonymous.

Thus, weak synonymy is the weakest of all synonym relations. In automatically finding lemmas that happen to fall in the same synset, we treat a synset as an equivalence class where the equivalence relation is weak synonymy. Note that our definition of weak synonymy is different than the near-synonymy given in Edmonds and Hirst [5] where the proximity is defined of over aspects such as style and indirectness. In our case, the proximity is restricted to occurrence in the dictionary definitions of each other.

To evaluate the performance of fully automatic synset construction, we experimented with the following two rules:

R_1: Slice the definition at the commas. A slice is a synonym candidate if it has an entry in the dictionary.

R_2: Slice the definition at the commas. Take as candidates all the slices that are to the right of the last slice that does not have dictionary entry.

R_1 is more inclusive than R_2. However, R_2 is more aligned with how the definitions are given in typical Turkish dictionaries that do not mark up the synonyms. In the CDT, usually a longer descriptive definition is given first, which is then followed by comma-separated synonyms or closely related terms. Of course, the descriptive part may also include commas. R_2 tries to eliminate such cases of false synonyms.

As an illustration, suppose that the definition of a sense is

$$w_1\ w_2\ w_3,\ w_4,\ w_5\ w_6,\ w_7,\ w_8,$$

where w_i are words. Further, for simplicity, suppose that only single words have dictionary entries.

R_1 generates w_4, w_7, and w_8 as synonym candidates, whereas R_2 generates only w_7 and w_8 since the expression $w_5\ w_6$ does not have an entry in the dictionary.

We used R_1 and R_2 and the weak synonymy relation to construct the synset graph and determine the synsets by finding the connected components of the resulting directed graph.

Figure 5 shows the distribution of synsets for the two rules R_1 and R_2 that we described earlier.

6 COMPARISON OF SYNSETS

Given a dictionary with one or more senses assigned to each lemma, constructing synsets corresponds to a clustering of the set of senses. Thus, different synset construction methods yield

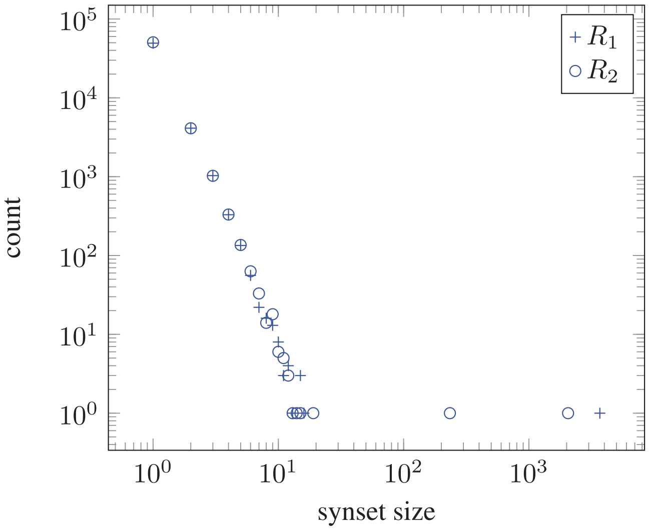


Fig. 5. Distribution of synset sizes for R_1 and R_2.

different clusterings of the set of senses. In this section, we evaluate the agreement between pairs of clusterings.

For measuring the distance between two clusterings, we use the variation of information that satisfies the properties of a metric [11]. Given a set A of size n and its two partitions $X_1, X_2, \ldots, X_k$ and $Y_1, Y_2, \ldots, Y_l$, the variation of information VI between the two is defined as

$$VI(X, Y) = -\sum_{i,j} r_{ij} \left[\log \frac{r_{ij}}{p_i} + \log \frac{r_{ij}}{q_j} \right],$$

where $p_i = \frac{|X_i|}{n}$, $q_j = \frac{|Y_j|}{n}$ and $r_{ij} = \frac{|X_i \cap Y_j|}{n}$.

Since the automatic constructions given in the previous section yields synsets of lemmas rather than those of senses, we make the comparisons over the sets of lemmas. However, this reduction poses a problem for manual synsets. Two distinct synsets might contain a common lemma, and when we consider only the lemmas, the two synsets would need to be joined, yielding an unnaturally large synset. To be able to discard such cases, for manual construction we considered only the lemmas with a single sense. Thus, synsets of senses and synsets of lemmas become the same.

We have the following six distinct methods of synset constructions. and each one yields a different partitioning:

MS: We use the set of manually determined pairs given in Table 2 as edges and find the connected components of the resulting sense graph. We eliminate the pairs where one of the lemmas has more than one sense in the CDT.

BS: We use the synsets in BalkaNet where each lemma has only a single sense.

ASR1: We first prune the dictionary by discarding the lemmas with more than one sense. We construct the synsets by automatically detecting weak synonym pairs under rule R_1.

ASR2: The same as ASR1, except we use R_2 instead of R_1.

AMR1: The same as ASR1, except we do not prune the dictionary but keep the lemmas with multiple senses.

AMR2: The same as AMR1, except we use R_2 instead of R_1.

Table 5. Variation of Information Among Different Synset Construction Methods

	Synset Construction Method				
	BS	ASR1	ASR2	AMR1	AMR2
MS	0.138	0.066	0.100	0.527	0.326
BS		0.134	0.161	**0.607**	0.384
ASR1			**0.030**	0.241	0.155
ASR2				0.265	0.158
AMR1					0.272

Table 5 gives the variation of information between the pairs of synset partitionings constructed using the preceding methods. Note that not every method works with the same set of lemmas. This is especially apparent for the case of BalkaNet, where the set of lemmas is considerably smaller than the other methods. To make a fair comparison, we measured the variation of information over the set of lemmas that is common to both methods in a pairwise comparison. Thus, in a comparison, if a particular lemma occurs in a synset in one partitioning but not in the other, we remove the lemma from the synset.

Table 5 highlights a couple of similarities and differences among construction methods. Among the automated methods, the variation distances seem to align them on a line as ASR2 < ASR1 < AMR2 < AMR1. Thus, ASR2 and ASR1 are quite similar since they confine their search within the set of lemmas that have a single sense. However, AMR2 and AMR1 are not as close. This is intuitively expected as when we consider multiple senses, determining that synonym candidates with comma splitting or right splitting tend to make a larger difference in the resulting synsets.

The same alignment can be observed when we compare BS and MS to automated methods. For both, the automated method that comes closest is ASR1.

Finally, we see that BS and MS are quite similar when projected onto the set of single-sensed lemmas appearing in BalkaNet.

7 CONCLUSION AND DISCUSSION

Constructing a WordNet is a manually intensive undertaking. In the present article, we presented a summary of our work on building a comprehensive WordNet for Turkish. Our manual annotation involved a total of nine human annotators over a period of 2 years.

In our WordNet construction, we mined a comprehensive dictionary of Turkish for synsets. We manually annotated the synsets twice, going over the disagreements for further reliability. We used clustering on the sense graph to find the final synsets.

For Turkish, WordNet construction is made more difficult by the lack of structured lexical resources. The most authoritative resource for Turkish lexicon is the official CDT published by the TLI. As we discussed in the preceding sections, the CDT has some lexicographical issues. The most acute of these for a WordNet study is the fuzzy boundaries among the senses of a lemma. Although a certain level of imprecision is often expected in lexicography, its level in the CDT makes its use in an NLP pipeline difficult. In our study, we used the CDT as it is while also noting the areas where it can be improved in a further study at a more fundamental level. Such a study would best start by collapsing some close senses into a single sense.

In our work, human annotators are presented with synonym candidates automatically mined from a monolingual dictionary. Obviously, one cannot expect an unstructured general dictionary to be comprehensive in listing synonym candidates. As a further study, one can imagine mining a large corpus for synonym candidates using contextual clues. In such an analysis for Turkish, context should be made canonical by stripping inflectional morphemes off the lemmas.

For the next stage of our WordNet construction, we will devise methods to automatically break down the huge synsets using contextual clues both from the definitions and corpora. We will still need to verify the resulting components through human annotators. Such a study will also provide us with further guidance on how to structure a canonical dictionary of Turkish.

The current version of KeNet is publicly available for download [6].

ACKNOWLEDGMENTS

We would like to thank the annotators, Onur Evren Alan, Tuğberk Arköse, Can Atay, Alişen Erdoğan, Melike Geçer, Can Kavaloğlu, Gizem Kısa, Berke Özenç, and Emre Şaşmaz.

We would like to thank the anonymous referees for their comments and suggestions which greatly improved an earlier version of this paper.

REFERENCES

[1] Daniil Alexeyevsky and Anastasiya V. Temchenko. 2016. WSD in monolingual dictionaries for Russian WordNet. In *Proceedings of the 8th Global WordNet Conference (GWC'16).*

[2] Global WordNet Association. 2017. Wordnets in the World. Retrieved March 23, 2018, from http://globalwordnet.org/wordnets-in-the-world/.

[3] Orhan Bilgin, Özlem Çetinoğlu, and Kemal Oflazer. 2004. Building a WordNet for Turkish. *Romanian Journal of Information Science and Technology* 7, 1–2, 163–172.

[4] William Black, Sabri Elkateb, Horacio Rodriguez, Musa Alkhalifa, Piek Vossen, Adam Pease, and Christiane Fellbaum. 2006. Introducing the Arabic WordNet project. In *Proceedings of the 3rd International WordNet Conference.* 295–300.

[5] Philip Edmonds and Graeme Hirst. 2002. Near-synonymy and lexical choice. *Computational Linguistics* 28, 2, 105–144. DOI: http://dx.doi.org/10.1162/089120102760173625

[6] Razieh Ehsani, Ercan Solak, and Olcay T. Yıldız. 2017. KeNet. Retrieved March 23, 2018, from http://haydut.isikun.edu.tr/kenet.html.

[7] Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database.* MIT Press, Cambridge, MA.

[8] Sangno Lee, Soon-Young Huh, and Ronald D. McNiel. 2008. Automatic generation of concept hierarchies using WordNet. *Expert Systems With Applications* 35, 3, 1132–1144.

[9] Cheng Hua Li, Ju Cheng Yang, and Soon Cheol Park. 2012. Text categorization algorithms using semantic approaches, corpus-based thesaurus and WordNet. *Expert Systems With Applications* 39, 1, 765–772.

[10] Krister Lindén, Jyrki Niemi, and Mirka Hyvärinen. 2012. *Extending and updating the Finnish Wordnet.* In *Shall We Play the Festschrift Game?* Springer, Berlin, Germany, 67–98. DOI: http://dx.doi.org/10.1007/978-3-642-30773-7_7

[11] Marina Meilă. 2003. Comparing clusterings by the variation of information. In *Learning Theory and Kernel Machines.* Springer, Berlin, Germany, 173–187.

[12] George A. Miller. 1995. WordNet: A lexical database for English. *Communications of the ACM* 38, 11, 39–41.

[13] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller. 1990. Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography* 3, 4, 235–244.

[14] Martha Palmer, Hoa Trang Dang, and Christiane Fellbaum. 2007. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering* 13, 2, 137–163.

[15] Maciej Piasecki, Stan Szpakowicz, Marek Maziarz, and Ewa Rudnicka. 2016. plWordNet 3.0 almost there. In *Proceedings of the 8th Global WordNet Conference (GWC'16).*

[16] Oxford University Press. 2017. Oxford Living Dictionaries. Retrieved March 23, 2018, from https://en.oxforddictionaries.com.

[17] Satu Elisa Schaeffer. 2007. Survey: Graph clustering. *Computer Science Review* 1, 1, 27–64. DOI: http://dx.doi.org/10.1016/j.cosrev.2007.05.001

[18] Rion Snow, Sushant Prakash, Daniel Jurafsky, and Andrew Y. Ng. 2007. Learning to merge word senses. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL'07).* 1005–1014.

[19] Dan Tufis, Dan Cristea, and Sofia Stamou. 2004. BalkaNet: Aims, methods, results and perspectives. A general overview. *Romanian Journal of Information Science and Technology* 7, 1–2, 9–43.

[20] Piek Vossen. 1997. EuroWordNet: A multilingual database for information retrieval. In *Proceedings of the DELOS Workshop on Cross-Language Information Retrieval*. 5–7.

[21] Tingting Wei, Yonghe Lu, Huiyou Chang, Qiang Zhou, and Xianyu Bao. 2015. A semantic approach for text clustering using WordNet and lexical chains. *Expert Systems With Applications* 42, 4, 2264–2275.

[22] George Kingsley Zipf. 1935. *The Psychobiology of Language*. Houghton-Mifflin, New York, NY.

Received July 2017; revised January 2018; accepted February 2018