Integrating WordNet and FrameNet
using a knowledge-based Word Sense Disambiguation algorithm

Egoitz Laparra and German Rigau
IXA group. University of the Basque Country, Donostia
{*egoitz.laparra,german.rigau*}@ehu.com

Abstract

This paper presents a novel automatic approach to partially integrate FrameNet and WordNet. In that way we expect to extend FrameNet coverage, to enrich WordNet with frame semantic information and possibly to extend FrameNet to languages other than English. The method uses a knowledge-based Word Sense Disambiguation algorithm for linking FrameNet lexical units to WordNet synsets. Specifically, we exploit a graph-based Word Sense Disambiguation algorithm that uses a large-scale knowledge-base derived from WordNet. We have developed and tested four additional versions of this algorithm showing a substantial improvement over previous results.

1 Introduction

Predicate models such as FrameNet [6], VerbNet [16] or PropBank [23] are core resources in most advanced NLP tasks, such as Question Answering, Textual Entailment or Information Extraction. Most of the systems with Natural Language Understanding capabilities require a large and precise amount of semantic knowledge at the predicate-argument level. This type of knowledge allows to identify the underlying typical participants of a particular event independently of its realization in the text. Thus, using these models, different linguistic phenomena expressing the same event, such as active/passive transformations, verb alternations and nominalizations can be harmonized into a common semantic representation. In fact, lately, several systems have been developed for shallow semantic parsing and semantic role labeling using these resources [11], [26], [14].

However, building large and rich enough predicate models for broad–coverage semantic processing takes a great deal of expensive manual effort involving large research groups during long periods of development. Thus, the coverage of currently available predicate-argument resources is still unsatisfactory. For example, [7] or [25] indicate the limited coverage of FrameNet as one of the main problems of this resource. In fact, FrameNet1.3 covers around 10,000 lexical-units while for instance, WordNet3.0 contains more than 150,000 words. Furthermore, the same effort should be invested for each different language [27]. Following the line of previous works [26], [7], [15], [24], [8], [29], we empirically study a novel approach to partially integrate FrameNet [6] and WordNet [12]. The method relies on the use of a knowledge-based Word Sense Disambiguation (WSD) algorithm that uses a large-scale graph of concepts derived from WordNet [12] and eXtented Word-Net [19]. The WSD algorithm is applied to semantically coherent groupings of words belonging to the same frame. In that way we expect to extend the coverage of FrameNet (by including from WordNet closely related concepts), to enrich WordNet with frame semantic information (by porting frame information to WordNet) and possibly to extend FrameNet to languages other than English (by exploiting local wordnets aligned to the English WordNet).

WordNet[1] [12] (hereinafter WN) is by far the most widely-used knowledge base. In fact, WN is being used world-wide for anchoring different types of semantic knowledge including wordnets for languages other than English [4], domain knowledge [17] or ontologies like SUMO [22] or the EuroWordNet Top Concept Ontology [3]. It contains manually coded information about English nouns, verbs, adjectives and adverbs and is organized around the notion of a *synset*. A synset is a set of words with the same part-of-speech that can be interchanged in a certain context. For example, <*student*, *pupil*, *educatee*> form a synset because they can be used to refer to the same concept. A synset is often further described by a gloss, in this case: "a learner who is enrolled in an educational institution" and by explicit semantic relations to other synsets. Each synset represents a concept which is related to other concepts by means of a large number of semantic relationships, including hypernymy/hyponymy, meronymy/holonymy, antonymy, entailment, etc.

FrameNet[2] [6] is a very rich semantic resource that contains descriptions and corpus annotations of English words following the paradigm of Frame Semantics [13]. In frame semantics, a Frame corresponds to a scenario that involves the interaction of a set of typical participants, playing a particular role in the scenario. FrameNet groups words (lexical units, LUs hereinafter) into coherent semantic classes or frames, and each frame is further characterized by a list of participants (lexical elements, LEs, hereinafter). Different senses for a word are represented in FrameNet by assigning different frames.

Currently, FrameNet represents more than 10,000 LUs and 825 frames. More than 6,100 of these LUs also provide linguistically annotated corpus examples. However, only 722 frames have associated a LU. From those, only 9,360 LUs[3] where recognized by WN (out of 92%) corresponding to only 708 frames.

LUs of a frame can be nouns, verbs, adjectives and adverbs representing a coherent and closely related set of meanings that can be viewed as a small semantic field. For example, the frame EDUCATION_TEACHING contains LUs referring to the teaching activity and their par-

---

[1] http://wordnet.princeton.edu/
[2] http://framenet.icsi.berkeley.edu/
[3] Word-frame pairs

ticipants. It is evoked by LUs like *student.n*, *teacher.n*, *learn.v*, *instruct.v*, *study.v*, etc. The frame also defines core semantic roles (or FEs) such as STUDENT, SUBJECT or TEACHER that are semantic participants of the frame and their corresponding LUs (see example below).

[Bernard Lansky]$_{STUDENT}$ *studied* [the piano]$_{SUBJECT}$ [with Peter Wallfisch]$_{TEACHER}$.

Table 1 presents the result of the our WSD process on some LUs of the Frame EDUCATION_TEACHING. We also include the polysemy degree of each word (#senses) and the definition (Gloss) of the sense (Synset) selected by the algorithm.

The contribution of this new resource is threefold[4]. First, we extend the coverage of FrameNet. For instance, the frame EDUCATION_TEACHING only considers *instruct.v* and *instruction.n*, but not *instructor.n* which is a synonym in WN of the LU *teacher.n*. Second, we can extend the coverage of semantic relations in WN. For instance, in WN there is no a semantic relation connecting <*student*, *pupil*, *educatee*> and < *teacher*, *instructor*> directly. Third, we can also automatically extend FrameNet to languages other than English by exploiting local wordnets aligned to the English WN. For instance, the Spanish synset aligned to <*student*, *pupil*, *educatee*> is <*alumno*, *estudiante*> and the Italian one is <*allievo*, *alunno*, *studente*>. Furthermore, we can also transport to the disambiguated LUs the knowledge currently available from other semantic resources associated to WN such as SUMO [22], WN Domains [17], etc. For instance, now the LU corresponding to *student.n* can also have associated the SUMO label *SocialRole* and its corresponding logical axioms, and the WN Domains *school* and *university*.

The paper is organized as follows. After this short introduction, in section 2 we present the graph-based Word Sense Disambiguation algorithm and the four additional versions studied in this work. The evaluation framework and the results obtained by the different algorithms are presented and analyzed in section 3, and finally, in section 4, we draw some final conclusions and outline future work.

2 SSI algorithms

Structural Semantic Interconnections (SSI) is a knowledge-based iterative approach to Word Sense Disambiguation [21]. The original SSI algorithm is very simple and consists of an initialization step and a set of iterative steps.

Given W, an ordered list of words to be disambiguated, the SSI algorithm performs as follows. During the initialization step, all monosemous words are included into the set I of already interpreted words, and the polysemous words are included in P (all of them pending to be disambiguated). At each step, the set I is used to disambiguate one word of P, selecting the word sense which is closer to the set I of already disambiguated words. Once a sense is disambiguated, the word sense is removed from P and included into I. The algorithm finishes when no more pending words remain in P.

In order to measure the proximity of one synset (of the word to be disambiguated at each step) to a set of synsets

[4] Available at `http://adimen.si.ehu.es/WordFrameNet`

Algorithm 1 SSI-Dijkstra algorithm

Function SSI-Dijkstra (T: list of terms)
$(I, P) := InitialInterpretation(T)$
for *each* $\{p \in P\}$ **do**
  $s := BestSense(p, I, \emptyset)$
  $I := I \cup \{s\}$
end for
Function InitialInterpretation (T: list of terms)
$(I, P) := SelectMonosemous(T)$
Function SelectMonosemous (T: list of terms)
$I := \emptyset$
for *each* $\{t \in T\}$ **do**
  if t *is monosemous* **then**
    $I := I \cup \{the \ unique \ sense \ of \ t\}$
  else
    $P := P \cup \{t\}$
  end if
end for
Function BestSense (t: term, I: list of senses, P: list of terms)
$BestSense := \emptyset$
$MinDistance := 0$
for *each* $\{sense \ s \ \in t\}$ **do**
  $d := MinDistanceS(s, I)$
  if $MinDistance = 0 \ or \ d < MinDistance$ **then**
    $BestSense := s$
    $MinDistance := d$
  end if
end for
Function MinDistance (s: sense, I: list of senses)
$d := 0$
for *each* $\{sense \ s' \in I\}$ **do**
  $d := d + DijkstraShortestPath(s, s')$
end for

(those word senses already interpreted in I), the original SSI uses an in-house knowledge base derived semi-automatically which integrates a variety of online resources [20]. This very rich knowledge-base is used to calculate graph distances between synsets. In order to avoid the exponential explosion of possibilities, not all paths are considered. They used a context-free grammar of relations trained on SemCor to filter-out inappropriate paths and to provide weights to the appropriate paths.

Instead, we used a version of the SSI algorithm called SSI-Dijkstra [9] (see algorithm 1. SSI-Dijkstra uses the Dijkstra algorithm to obtain the shortest path distance between a node and some other nodes of the whole graph. The Dijkstra algorithm is a greedy algorithm that computes the shortest path distance between one node an the rest of nodes of a graph. BoostGraph[5] library can be used to compute very efficiently the shortest distance between any two given nodes on very large graphs. As [9], we also use already available knowledge resources to build a very large connected graph with 99,635 nodes (synsets) and 636,077 edges (the set of direct relations between synsets gathered from WN[6][12] and eXtended WN[7] [19]. For building this graph we used WN version 1.6 and the semantic relations appearing between synsets and disambiguated glosses of WN 1.7. To map the relations appearing in eXtended WN to WN version 1.6 we used the automatic WN Mappings[8] [10]. On that graph, SSI-Dijkstra computes several times

[5] http://www.boost.org/doc/libs/1_35_0/libs/graph/doc/index.html
[6] http://wordnet.princeton.edu
[7] http://xwn.hlt.utdallas.edu
[8] http://www.lsi.upc.es/ nlp/tools/mapping.html

Lexical Unit	synset	#senses	Gloss
education.n	00567704-n	2	"activities that impart knowledge"
teacher.n	07632177-n	2	"a person whose occupation is teaching"
instruct.v	00562446-v	3	"impart skills or knowledge"
study.v	00410381-v	6	"be a student; follow a course of study; be enrolled at an institute of learning"
student.n	07617015-n	2	"a learner who is enrolled in an educational institution"
pupil.n	07617015-n	3	"a learner who is enrolled in an educational institution"

Table 1: *Partial result of the WSD process of the LUs of the frame EDUCATION_TEACHING*

the Dijkstra algorithm.

SSI-Dijkstra has very interesting properties. For instance, as the Dijkstra algorithm always provides the minimum distance between two synsets, the SSI-Dijkstra algorithm always provides an answer being the minimum distance close or far. In contrast, the original SSI algorithm not always provides a path distance because it depends on a predefined grammar of semantic relations. In fact, the SSI-Dijkstra algorithm compares the distances between the synsets of a word and all the synsets already interpreted in I. At each step, the SSI-Dijkstra algorithm selects the synset which is closer to I (the set of already interpreted words).

Previously, the SSI-Dijkstra algorithm have been used for constructing KnowNets [9]. KnowNets are very large knowledge bases, which have been acquired by semantically disambiguating the Topic Signatures obtained from the web [1]. Basically, the method uses SSI-Dijkstra to assign the most appropriate senses to large sets of ordered topic words (for instance, *underclassman*, *overachiever*, *seminarian*, *college*, etc.) associated to a particular synset (for instance, *pupil#n#1*).

Initially, the list I of interpreted words should include the senses of the monosemous words in W, or a fixed set of word senses. Note that when disambiguating a Topic Signature associated to a particular synset, the list I always includes since the beginning of the process at least the sense of the Topic Signature (in our example *pupil#n#1*) and the rest of monosemous words of W. However, many frames only group polysemous LUs. In fact, a total of 190 frames (out of 26%) only have polysemous LUs. Thus, SSI-Dijkstra provides no results when there are no monosemous terms in W. In this case, before applying SSI, the set of the LUs corresponding to a frame (the words included in W) have been ordered by polysemy degree. That is, the less polysemous words in W are processed first.

Obviously, if no monosemous words are found, we can adapt the SSI algorithm to make an initial guess based on the most probable sense of the less ambiguous word of W. For this reason we implemented two different versions of the basic SSI-Dijkstra algorithm: **SSI-Dijkstra-FirstSenses-I** (hereinafter FSI) and **SSI-Dijkstra-AllSenses-I** (hereinafter ASI). Thus, these two versions perform as SSI-Dijkstra when W contains monosemous terms, but differently when W contains only polysemous words. In fact, FSI and ASI always provide an interpretation of W.

While FSI includes in I the sense having minimal cumulated distance to the first senses of the rest of words in W, ASI includes in I the sense having minimal cumulated distance to the all the senses of the rest of words in W. The rationale behind the FSI algorithm is that the most frequent sense for a word, according to the WN sense ranking is very

competitive in WSD tasks, and it is extremely hard to improve upon even slightly [18]. Thus, this algorithm expects that the first sense in WN will be correct for most of the words in W. Regarding ASI, this algorithm expects that the words in W (corresponding to a very close semantic field) will establish many close path connections between different synsets of the same word (because of the fine-grained sense distinction of WN).

At each step, both the original SSI and also the SSI-Dijkstra algorithms only consider the set I of already interpreted words to disambiguate the next word of P. That is, the remaining words of P are not used in the disambiguation process. In fact, the words in P are still not disambiguated and can introduce noise in the process. However, the knowledge remaining in P can also help the process. In order to test the contribution of the remaining words in P in the disambiguation process, we also developed two more versions of the basic SSI-Dijkstra algorithm. **SSI-Dijkstra-FirstSenses-P** (hereinafter FSP) and **SSI-Dijkstra-AllSenses-P** (hereinafter ASP). When a word is being disambiguated, these two versions consider the set I of already interpreted words of W and also the rest of words remaining in P. That is, at each step, the algorithm selects the word sense which is closer to the set I of already disambiguated words and the remaining words of P all together. While FSP selects the sense having minimal cumulated distance to I and the first senses of the words in P, ASP selects the sense having minimal cumulated distance to I and all the senses of the words in P.

3 Experiments

We have evaluated the performance of the different versions of the SSI algorithm using the same data set used by [28] and [29]. This data set consists of a total of 372 LUs corresponding to 372 different frames from FrameNet1.3 (one LU per frame). Each LUs have been manually annotated with the corresponding WN 1.6 synset. This Gold Standard includes 9 frames (5 verbs and 4 nouns) with only one LU (the one that has been sense annotated). Obviously, for these cases, our approach will produce no results since no context words can be used to help the disambiguation process[9]. Table 2 presents the main characteristics of the datasets we used in this work. In this table, *FN* stands for FrameNet[10], *GS* for the Gold-Standard, *mono* for those Gold-Standard frames having at least one monosemous LU and *poly* for those Gold-Standard frames having only polysemous LUs. The table shows for each dataset, the number of frames and the average distribution per frame of

---

[9] In fact, FrameNet has 33 frames with only one LU, and 63 with only two.

[10] We removed frames with no LUs assigned or not present in WN

	FN	GS	mono	poly	10
#Frames	708	372	299	73	195
Nouns	5.87	7.90	9.35	1.95	13.58
Verbs	5.77	6.49	7.32	3.09	9.70
Adjectives	2.49	3.24	3.86	0.71	5.36
Other	0.11	0.14	0.14	0.09	0.24
Not in WN	1.07	1.30	1.51	0.42	2.13
Monosemous	4.40	5.79	7.20	0.00	9.87
Polysemous	8.77	10.68	11.96	5.42	16.88
#senses	3.64	3.45	3.28	5.64	3.63
Total	14.24	17.77	20.67	5.84	28.88

Table 2: *Number of frames and average distribution of words per frame of the different datasets*

each POS, the words not represented in WN, the number of monosemous and polysemous words, the polysemy degree and the total words. The number of words per frame in this Gold Standard seems to be higher than the average in FrameNet. This data set also has 73 frames having only polysemous LUs (20% of the total). That is, these frames do not have monosemous LUs. Possibly, because its small size (5.84 words on average).

Table 3 presents detailed results per Part-of-Speech (POS) of the performance of the different SSI algorithms in terms of Precision (P), Recall (R) and F1 measure (harmonic mean of recall and precision). In bold appear the best results for precision, recall and F1 measures. As baseline, we also include the performance measured on this data set of the most frequent sense according to the WN sense ranking. Remember that this baseline is very competitive in WSD tasks, and it is extremely hard to beat. However, all the different versions of the SSI-Dijkstra algorithm outperform the baseline. Only SSI-Dijkstra obtains lower recall for verbs because of its lower coverage. In fact, SSI-Dijkstra only provide answers for those frames having monosemous LUs, the SSI-Dijkstra variants provide answers for frames having at least two LUs (monosemous or polysemous) while the baseline always provides an answer.

As expected, the SSI algorithms present different performances according to the different POS. Also as expected, verbs seem to be more difficult than nouns and adjectives as reflected by both the results of the baseline and the SSI-Dijkstra algorithms. For nouns and adjectives, the best results are achieved by both FSI and ASI variants. Remember that these versions perform as SSI-Dijkstra on frames having monosemous LUs but performing an initial guess on frames having only polysemous LUs. While FSI makes an initial guess including in I the sense of the less polysemous word having minimal cumulated distance to the *first senses* of the rest of words in W, ASI makes an initial guess including in I the sense of the less polysemous word having minimal cumulated distance to *all* the senses of the rest of words in W. In fact, FSI and ASI behave differently than SSI-Dijkstra in the 73 frames having only polysemous LUs in the data set. Interestingly, the best results for verbs are achieved by FSP, not only on terms of F1 but also on precision. Remember that FSP always uses I and the first senses of the rest of words in P as context for the disambiguation. It seems that for verbs it is useful to consider not only the disambiguated words but also the most frequent senses of the rest of words being disambiguated. However, for nouns and adjectives the best precision is achieved by the original SSI-Dijkstra. This fact suggests the importance of hav-

ing monosemous or correctly disambiguated words in I at the beginning of the incremental disambiguation process, at least for nouns and adjectives.

To our knowledge, on the same dataset, the best results so far are the ones presented by [29]. They presented a novel machine learning approach reporting a Precision of 0.76, a Recall of 0.61 and an F measure of 0.68[11]. Note that these results are below the most-frequent sense according to the WN sense ranking (F1=0.69) and all versions of SSI-Dijkstra (F1 from 0.69 to 0.74).

In order to measure the contribution of the different SSI-Dijkstra versions on those frames having at least one monosemous LU, Table 4 presents detailed results per POS of its performance in terms of Precision (P), Recall (R) and F1 measure (F). Again, in bold appear the best results, and as a baseline, we again include the results measured on this data set of the most frequent sense according to the WN sense ranking. Obviously, FSI and ASI variants are not included since for frames having monosemous LUs both approaches obtain the same result as of the SSI-Dijkstra algorithm. Interestingly, when having monosemous LUs, all SSI algorithms obtain substantial improvements over the baseline, which is very high. Also interesting is that SSI-Dijkstra obtains the best results for nouns and adjectives while FSP obtains the best results for verbs.

In order to measure the contribution of the different SSI-Dijkstra versions on those 73 frames having only polysemous LUs, Table 5 presents detailed results per POS of its performance in terms of Precision (P), Recall (R) and F1 measure (F). Again, in bold appear the best results, and as a baseline, we again include the results measured on this data set of the most frequent sense according to the WN sense ranking. Obviously, the original SSI-Dijkstra is not included. For these subset of frames, the algorithms behave similarly as for the whole data set. In fact, as before, verbs seem to be more difficult than nouns and adjectives. However, according to the baseline, without monosemous LUs the task seems to be much more difficult. This is specially acute for nouns and verbs where the the first sense heuristic obtains accuracies of 58% and 48% respectively. The algorithms also present different performances according to the different POS. Again, the the best results are achieved by both FSI and ASI variants on nouns and adjectives, and FSP on verbs. However, in this data set only ASI slightly outperforms the baseline in precision and F1. Since these versions do not provide answers for frames having only one LU, the recall is below precision.

Although the set of frames having only polysemous LUs seems to be much more difficult than the set of frames having monosemous LUs, the results shown in tables 4 and 5 also suggest room for improving the SSI algorithms. In fact, not only for frames having no monosemous LUs, but also in general. For instance, for disambiguating verbs. These results suggest that possibly, a new version of the SSI-Dijkstra algorithm processing nouns and adjectives as FSI (or ASI) and verbs as FSP would clearly outperform the current versions. We expect for this new algorithm improved results also for nouns, verbs and adjectives, since the whole incremental disambiguation process will benefit from a better disambiguation of I. Possibly, during the incremental and iterative disambiguation process, a better disambiguation of verbs will improve the disambiguation

---

[11] In fact, both evaluations are slightly different since they perform 10-fold cross validation on the available data, while we provide results for the whole dataset.

	nouns			verbs			adjectives			all		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
wn-mfs	0.75	0.75	0.75	0.64	0.64	0.64	0.80	0.80	0.80	0.69	0.69	0.69
SSI-dijktra	**0.84**	0.65	0.73	0.70	0.56	0.62	**0.90**	0.82	0.86	**0.78**	0.63	0.69
FSI	0.80	**0.77**	**0.79**	0.66	0.65	0.65	0.89	**0.89**	**0.89**	0.74	**0.73**	0.73
ASI	0,80	**0,77**	**0,79**	0,67	0,65	0,66	0,89	**0,89**	**0,89**	0,75	**0,73**	**0,74**
FSP	0.75	0.73	0.74	**0.71**	**0.69**	**0.70**	0.79	0.79	0.79	0.73	0.72	0.72
ASP	0.72	0.69	0.70	0.68	0.66	0.67	0.75	0.75	0.75	0.70	0.69	0.69

Table 3: *Results of the different SSI algorithms*

	nouns			verbs			adjectives			all		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
wn-mfs	0.78	0.78	0,78	0,67	0,67	0,67	0,80	0,80	0,80	0,73	0,73	0,73
SSI-dijktra	**0,84**	**0,84**	**0,84**	0,70	0,70	0,70	**0,90**	**0,90**	**0,90**	**0,78**	**0,78**	**0,78**
FSP	0,80	0,78	0,79	**0,73**	**0,73**	**0,73**	0,80	0,80	0,80	0,76	0,76	0,76
ASP	0,78	0,78	0,78	0,71	0,71	0,71	0,76	0,76	0,76	0,74	0,74	0,74

Table 4: *Evaluation of frames with at least one monosemous word*

of nouns, and a better disambiguation of nouns will also improve the disambiguation of verbs and adjectives.

However, still remains unclear if the problem of frames having no monosemous LUs is because the lack of correctly disambiguated words in I, the small number of LUs per frame or its high polysemy degree. We expect to clarify this issue in future experiments and analysis.

Although the experimental setting is different, [8] also present a direct evaluation of their integration of WN and FrameNet for the LU induction task [5]. They apply a combination of knowledge and distributional based methods to carry out the mapping process. In order to aliviate their data sparseness problem, they reduced the whole dataset in two ways. First, they neglected LUs occurring less than 50 times in the British National Corpus. Second, they excluded frames having less than 10 LUs. This leaves them with 220 frames, involving 4,380 LUs. They focused the study of the quality of their automatic mapping on four frames (i.e. KILLING, PEOPLE_BY_AGE, STATEMENT and CLOTHING) with 306 LUs. On this dataset, they report a precision of 0.80, a recall of 0.79 and an F measure of 0.80. Unfortunately, they do not report detailed performances per POS nor baselines. Trying to be more representative of the whole resource, the dataset used in our study covers a large set of frames but only one LU per frame has been annotated. Obviously, the results of these four frames will not allow to make appropriate conclusions.

In order to establish a fair comparison with our evaluation framework, Table 6 also presents detailed results per POS of the performance of the SSI versions in terms of Precision (P), Recall (R) and F1 measure (F) on the 195 frames having at least 10 LUs[12]. Again, in bold appear the best results, and as a baseline, we again include the results measured on this reduced data set of the most frequent sense according to the WN sense ranking. Note that the average result for this baseline is the same as the one reported for the whole dataset although it presents a different behaviour depending on the POS. Regarding SSI algorithms, they behave similarly as with the whole dataset (better precision for SSI-Dijkstra, better performance for FSI and ASI on nouns and adjectives and FSP for verbs,

and better performance overall for FSI and ASI). Surprisingly, the different SSI algorithms only obtain for nouns better performances than with the whole dataset. Slightly worst results are obtained for verbs and adjectives. Possibly, the cause of this phenomena would be the different POS distribution per frame on this particular dataset. However, overall, the results improve with respect the complete Gold-Standard.

Although both approaches are not directly comparable due to the different evaluation dataset, our results seem to be very close to those reported by [8]. In fact, their dataset excluded low frequent LUs and was centered only on the LUs of four frames. Moreover, we applied a unique knowledge based approach. Furthermore, we expect even better results with the improved version of the SSI-Dijkstra using FSI for nouns and adjectives, and FSP for verbs.

4   Conclusions and future work

In this work, we have presented a novel approach to integrate FrameNet and WordNet. The method uses a knowledge based Word Sense Disambiguation (WSD) algorithm called SSI-Dijkstra for assigning the appropriate synset of WordNet to the semantically related Lexical Units of a given frame from FrameNet. This algorithm relies on the use of a large knowledge base derived from WordNet and eXtended WordNet. Since the original SSI-Dijkstra requires a set of monosemous or already interpreted words, we have devised, developed and empirically tested four different versions of this algorithm to deal with sets having only polysemous words. The resulting new algorithms obtain improved results over state-of-the-art.

As a result of this empirical study, we are currently developping a new version of the SSI-Dijkstra using FSI for nouns and adjectives, and FSP for verbs. We also plan to further extend the empirical evaluation with other available graph based algorithms that have been proved to be competitive in WSD such as UKB[13] [2].

Finally, using the same automatic approach, we also plan to disambiguate the Lexical Elements of a given frame.

---

[12] We did not remove unfrequent LUs

[13] http://ixa2.si.ehu.es/ukb/

	nouns			verbs			adjectives			all		
	P	R	F	P	R	F	P	R	F	P	R	F
wn-mfs	0.58	**0.58**	0.58	0.48	0.48	0.48	**0.80**	**0.80**	**0.80**	0.54	**0.54**	0.54
FSI	**0,64**	0,55	**0,59**	0,50	0,44	0,47	**0,80**	**0,80**	**0,80**	0,58	0,51	0,54
ASI	**0,64**	0,55	**0,59**	0,53	0,46	0,49	**0,80**	**0,80**	**0,80**	**0,59**	0,52	**0,55**
FSP	0,56	0,48	0,52	**0,59**	**0,51**	**0,55**	0,60	0,60	0,60	0,58	0,51	0,54
ASP	0,48	0,41	0,44	0,53	0,46	0,49	0,60	0,60	0,60	0,52	0,45	0,48

Table 5: *Results of the different SSI algorithms on frames having only polysemous LUs*

	nouns			verbs			adjectives			all		
	P	R	F	P	R	F	P	R	F	P	R	F
wn-mfs	0.76	0.76	0.76	0.61	0.61	0.61	0.76	0.76	0.76	0.69	0.69	0.69
SSI-dijktra	**0,86**	0,78	0,82	0,66	0,63	0,64	**0,88**	0,85	0,87	**0,77**	0,72	0,75
FSI	0,85	**0,85**	**0,85**	0,64	0,64	0,64	**0,88**	**0,88**	**0,88**	0,76	**0,76**	**0,76**
ASI	0,85	**0,85**	**0,85**	0,64	0,64	0,64	**0,88**	**0,88**	**0,88**	0,76	**0,76**	**0,76**
FSP	0,81	0,81	0,81	**0,67**	**0,67**	**0,67**	0,74	0,74	0,74	0,73	0,73	0,73
ASP	0,76	0,76	0,76	0,63	0,63	0,63	0,71	0,71	0,71	0,69	0,69	0,69

Table 6: *Results of the different SSI algorithms on frames having at least 10 LUs*
.

Thus, the resulting resource will also integrate the core semantic roles of FrameNet. For example, for the frame EDUCATION_TEACHING we will associate the appropriate WordNet synsets to the Lexical Elements STUDENT, SUBJECT or TEACHER.

References

[1] E. Agirre and O. L. de la Calle. Publicly available topic signatures for all wordnet nominal senses. In *Proceedings of LREC*, Lisbon, Portugal, 2004.

[2] E. Agirre and A. Soroa. Personalizing pagerank for word sense disambiguation. In *Proceedings of the 12th conference of the European chapter of the Association for Computational Linguistics (EACL-2009)*, Athens, Greece, April 2009. Eurpean Association for Computational Linguistics.

[3] J. Álvez, J. Atserias, J. Carrera, S. Climent, A. Oliver, and G. Rigau. Consistent annotation of eurowordnet with the top concept ontology. In *Proceedings of Fourth International WordNet Conference (GWC'08)*, 2008.

[4] J. Atserias, L. Villarejo, G. Rigau, E. Agirre, J. Carroll, B. Magnini, and P. Vossen. The meaning multilingual central repository. In *Proceedings of GWC*, Brno, Czech Republic, 2004.

[5] C. Baker, M. Ellsworth, and K. Erk. Semeval-2007 task 19: Frame semantic structure extraction. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 99–104, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[6] C. Baker, C. Fillmore, and J. Lowe. The berkeley framenet project. In *COLING/ACL'98*, Montreal, Canada, 1997.

[7] A. Burchardt, K. Erk, and A. Frank. A WordNet Detour to FrameNet. In *Proceedings of the GLDV 2005 GermaNet II Workshop*, pages 408–421, Bonn, Germany, 2005.

[8] D. D. Cao, D. Croce, M. Pennacchiotti, and R. Basili. Combining word sense and usage for modeling frame semantics. In *Proceedings of The Symposium on Semantics in Systems for Text Processing (STEP 2008)*, Venice, Italy, 2008.

[9] M. Cuadros and G. Rigau. Knownet: Building a large net of knowledge from the web. In *22nd International Conference on Computational Linguistics (COLING'08)*, Manchester, UK, 2008.

[10] J. Daudé, L. Padró, and G. Rigau. Validation and Tuning of Wordnet Mapping Techniques. In *Proceedings of RANLP*, Borovets, Bulgaria, 2003.

[11] K. Erk and S. Pado. A powerful and versatile xml format for representing role-semantic annotation. In *Proceedings of LREC-2004*, Lisbon, 2004.

[12] C. Fellbaum, editor. *WordNet. An Electronic Lexical Database*. The MIT Press, 1998.

[13] C. J. Fillmore. Frame semantics and the nature of language. In *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, volume 280, pages 20–32, New York, 1976.

[14] A.-M. Giuglea and A. Moschitti. Semantic role labeling via framenet, verbnet and propbank. In *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 929–936, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[15] R. Johansson and P. Nugues. Using WordNet to extend FrameNet coverage. In *Proceedings of the Workshop on Building Frame-semantic Resources for Scandinavian and Baltic Languages, at NODALIDA*, Tartu, Estonia, May 24 2007.

[16] K. Kipper. *VerbNet: A broad-coverage, comprehensive verb lexicon*. PhD thesis, University of Pennsylvania, 2005.

[17] B. Magnini and G. Cavaglià. Integrating subject field codes into wordnet. In *Proceedings of LREC*, Athens. Greece, 2000.

[18] D. McCarthy, R. Koeling, J. Weeds, and J. Carroll. Finding predominant senses in untagged text. In *Proceedings of ACL*, pages 280–297, 2004.

[19] R. Mihalcea and D. Moldovan. extended wordnet: Progress report. In *Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*, Pittsburgh, PA, 2001.

[20] R. Navigli. Semi-automatic extension of large-scale linguistic knowledge bases. In *Proc. of 18th FLAIRS International Conference (FLAIRS)*, Clearwater Beach, Florida, 2005.

[21] R. Navigli and P. Velardi. Structural semantic interconnections: a knowledge-based approach to word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 27(7):1063–1074, 2005.

[22] I. Niles and A. Pease. Towards a standard upper ontology. In *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, pages 17–19. Chris Welty and Barry Smith, eds, 2001.

[23] M. Palmer, D. Gildea, and P. Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106, March 2005.

[24] M. Pennacchiotti, D. D. Cao, R. Basili, D. Croce, and M. Roth. Automatic induction of FrameNet lexical units. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*, Honolulu, Hawaii, USA, 2008.

[25] D. Shen and M. Lapata. Using semantic roles to improve question answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 12–21, 2007.

[26] L. Shi and R. Mihalcea. Putting pieces together: Combining framenet, verbnet and wordnet for robust semantic parsing. In *Proceedings of CICLing*, Mexico, 2005.

[27] C. Subirats and M. R. Petruck. Surprise: Spanish framenet! In *Proceedings of the International Congress of Linguists*, Praga, 2003.

[28] S. Tonelli and E. Pianta. A novel approach to mapping framenet lexical units to wordnet synsets. In *Proceedings of IWCS-8*, Tilburg, The Netherlands, 2009.

[29] S. Tonelli and D. Pighin. New features for framenet - wordnet mapping. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL'09)*, Boulder, CO, USA, 2009.