

Augmenting WordNet-based Inference with Argument Mapping

Idan Szpektor

Department of Computer Science
Bar-Ilan University
Ramat Gan, Israel
szpekti@cs.biu.ac.il

Ido Dagan

Department of Computer Science
Bar-Ilan University
Ramat Gan, Israel
dagan@cs.biu.ac.il

Abstract

WordNet is a useful resource for lexical inference in applications. Inference over predicates, however, often requires a change in argument positions, which is not specified in WordNet. We propose a novel framework for augmenting WordNet-based inferences over predicates with corresponding argument mappings. We further present a concrete implementation of this framework, which yields substantial improvement to WordNet-based inference.

1 Introduction

WordNet (Miller, 1995), a manually constructed lexical database, is probably the mostly used resource for lexical inference in NLP tasks, such as Question Answering (QA), Information Extraction (IE), Information Retrieval and Textual Entailment (RTE) (Moldovan and Mihalcea, 2000; Pasca and Harabagiu, 2001; Bar-Haim et al., 2006; Giampiccolo et al., 2007).

Inference using WordNet typically involves lexical substitutions for words in text based on WordNet relations, a process known as lexical chains (Barzilay and Elhadad, 1997; Moldovan and Novischi, 2002). For example, the answer to “*From which country was Louisiana acquired?*” can be inferred from “*The United States bought up Louisiana from France*” using the chains ‘France \Rightarrow European country \Rightarrow country’ and ‘buy up \Rightarrow buy \Rightarrow acquire’.

When performing inference between predicates there is an additional complexity on top of lexical substitution: the syntactic relationship between the predicate and its arguments may change

as well. For example, ‘ X buy Y for $Z \Rightarrow X$ pay Z for Y ’.

Currently, argument mappings are not specified for WordNet’s relations. Therefore, correct WordNet inference chains over predicates can be performed only for substitution relations (mainly synonyms and hypernyms, e.g. ‘buy \Rightarrow acquire’), for which argument positions do not change. Other relation types that may be used for inference cannot be utilized when the predicate arguments need to be traced as well. Examples include the WordNet ‘entailment’ relation (e.g. ‘buy \Rightarrow pay’) and relations between morphologically derived words (e.g. ‘acquire \Leftrightarrow acquisition’).

Our goal is to obtain argument mappings for WordNet relations that are often used for inference. In this paper we address several prominent WordNet relations, including verb-noun derivations and the verb-verb ‘entailment’ and ‘cause’ relations, referred henceforth as *inferential relations*. Under the Textual Entailment paradigm, all these relations can be viewed as expressing entailment. Accordingly, we propose a novel framework, called *Argument-mapped WordNet* (*AmWN*), that represents argument mappings for inferential relations as entailment rules. These rules are augmented with subcategorization frames and functional roles, which are proposed as a generally-needed extension for predicative entailment rules.

Following our new representation scheme, we present a concrete implementation of AmWN for a large number of WordNet’s relations. The mappings for these relations are populated by combining information from manual and corpus-based resources, which provides broader coverage compared to prior work and more accurate mappings. Table 1 shows typical inference chains obtained

Rule Chains
shopping:n of X_{obj} \Rightarrow buying:n of X_{obj} \Rightarrow buy:v X_{obj} \Rightarrow pay:v for X_{mod}
vote:v on X_{mod} \Rightarrow decide:v on X_{mod} \Rightarrow debate:v X_{obj}
X_{obj} 's sentence:n \Rightarrow condemn:v X_{obj} \Rightarrow convict:v X_{obj} \Rightarrow X_{obj} 's conviction:n
$X_{ind-obj}$'s teacher:n \Rightarrow teach:v to $X_{ind-obj}$ \Rightarrow X_{subj} learn:v

Table 1: Examples for inference chains obtained using AmWN. Arguments are subscripted with functional roles, e.g. subject (subj) and indirect-object (ind-obj). For brevity, predicate frames are omitted.

using our implementation.

To further improve WordNet-based inference for NLP applications, we address the phenomena of rare WordNet senses. Rules generated for such senses might hurt inference accuracy since they are often applied incorrectly to texts when matched against inappropriate, but more frequent senses of the rule words. Since word sense disambiguation (WSD) solutions are typically not sufficiently robust yet, most applications do not currently apply WSD methods. Hence, we propose to optionally filter out such rules using a novel corpus-based validation algorithm.

We tested both WordNet and AmWN on a test set derived from a standard IE benchmark. The results show that AmWN substantially improves WordNet-based inference in terms of both recall and precision¹.

2 Argument-Mapping Entailment Rules

In our framework we represent argument mappings for inferential relations between predicates through an extension of entailment rules over syntactic representations. As defined in earlier works, an *entailment rule* specifies an inference relation between an entailing template and an entailed template, where *templates* are parse subtrees with argument variables (Szpektor and Dagan, 2008). For example, ‘ $X \xleftarrow{subj} \text{buy} \xrightarrow{obj} Y$ ’ \Rightarrow ‘ $X \xleftarrow{subj} \text{pay} \xrightarrow{prep-for} Y$ ’.

When a rule is applied to a text, a new consequent is inferred by instantiating the entailed template variables with the argument instantiations of the entailing template in the text. In our example, “*IBM paid for Cognos*” can be inferred from “*IBM bought Cognos*”. This way, the syntactic structure of the rule templates specifies the required argument positions for correct argument mapping.

However, representing entailment rule structure only by syntactic argument positions is insufficient for predicative rules. Correct argument mapping

depends also on the specific *syntactic functional roles* of the arguments (subject, object etc.) and on the suitable *subcategorization frame (frame)* for the predicate mention - a set of functional roles that a predicate may occur with. For example, ‘ X 's buyout \Rightarrow buy X ’ is incorrectly applied to “*IBM's buyout of Cognos*” if roles are ignored, since ‘IBM’ plays the subject role while X needs to be an object.

Seeking to address this issue, we were inspired by the Nomlex database (Macleod et al., 1998) (see Section 3.2.1) and explicitly specify argument mapping for each frame and functional role. As in Nomlex, we avoid the use of semantic roles and stick to the syntactic level, augmenting the representation of templates with: (a) a syntactic functional role for each argument; (b) the valid predicate frame for this template mentions. We note that such functional roles typically coincide with dependency relations of the verbal form. A rule example is ‘ $X_{subj} \text{break}_{\{intrans\}} \Rightarrow \text{damage}_{\{trans\}} X_{obj}$ ’². More examples are shown in Table 1.

Unlike Nomlex records, our templates can be partial: they may contain only some of the possible predicate arguments, e.g. ‘ $\text{buy}_{\{trans\}} X_{obj}$ ’, where the subject, included in the frame, is omitted. Partial templates are necessary for matching predicate occurrences that include only some of the possible arguments, as in “*Cognos was bought yesterday*”. Additionally, some resources, such as automatic rule learning methods (Lin and Pantel, 2001; Sekine, 2005), can provide only partial argument information, and we would want to represent such knowledge as well.

In our framework we follow (Szpektor and Dagan, 2008) and use only rules between *unary templates*, containing a single argument. Such templates can describe any argument mapping by de-

²Functional roles are denoted by subscripts of the arguments and frames by subscripts of the predicate. We shorthand *trans* for transitive frame $\{subject, object\}$ and *intrans* for intransitive $\{subject\}$. For brevity, we will not show all template information when examples are self explanatory.

¹We plan to make our AmWN publicly available.

composing templates with several arguments into unary ones, while preserving the specification of the subcategorization frame.

To apply a rule, the entailing template must be first matched in the text, which includes matching the template’s syntactic dependency structure, functional roles, and frame. Such procedure requires texts to be annotated with these types of information. This can be reasonably performed with existing tools and resources, as described for our own text processing in Section 4.

Explicitly matching frames and functional roles in rules avoids incorrect rule applications. For example, ‘ X_{obj} ’s buyout \Rightarrow buy X_{obj} ’ would be applied only to “*Cognos’s buyout by IBM*” following proper role annotation of the text, but not to “*IBM’s buyout of Cognos*”. As another example, ‘ X_{subj} break_{intrans} \Rightarrow damage_{trans} X_{obj} ’ would be applied only to the intransitive occurrence of ‘break’, e.g. “*The vase broke*”, but not to “*John broke the vase*”.

Ambiguous cases may occur during annotation. For example, the role of ‘John’ in “*John’s invitation was well intended*” could be either subject or object. Such recognized ambiguities should be left unannotated, blocking incorrect rule application.

3 Argument Mapping for WordNet

Following our extension of entailment rules, we present *Argument-mapped WordNet (AmWN)*, a framework for extending WordNet’s inferential relations with argument mapping at the syntactic representation level.

3.1 Argument Mapping Representation

The AmWN structure follows that of WordNet: a directed graph whose nodes are WordNet synsets and edges are relations between synsets. Since we focus on entailment between predicates, we include only predicative synsets: all verb synsets and noun synsets identified as predicates (see Section 3.2). In addition, only WordNet relations that correspond to some type of entailment are considered, as detailed in Section 3.2.

In our framework, different subcategorization frames are treated as having different “meanings”, since different frames may correspond to different entailment rules. Each WordNet synset is split into several nodes, one for each of its frames. We take frame descriptions for verbs from WordNet³.

³We also tried using VerbNet (Kipper et al., 2000), with-

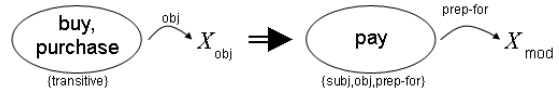


Figure 1: A description of ‘buy/purchase $X \Rightarrow$ pay for X ’ as a mapping edge in AmWN.

Since WordNet does not provide frames for noun predicates, these are taken from Nomlex-plus (see Section 3.2).

There are two types of graph edges that represent entailment rules between nodes: mapping edges and substitution edges. Mapping edges specify entailment rules that require argument mapping, where the entailing and entailed template predicates are replaced by synsets. Thus, an edge represents all rules between entailing and entailed synset members, as in Figure 1.

Substitution edges connect pairs of predicates, of the same part-of-speech, which preserve argument positions in inference. This is analogous to how WordNet may be currently used for inference via the *synonym* and *hypernym* relations. Unlike WordNet, substitution edges in AmWN may connect only nodes that have the same subcategorization frame.

AmWN is utilized by generating rule chains for a given input unary template. First, starting nodes that match the input predicate are selected. Then, rules are generated by traversing either incoming or outgoing graph edges transitively, depending on the entailment direction requested. Specific synset-ids, if known, may also be added to the input to constrain the relevant starting nodes for the input predicate. Table 1 shows examples of rule chains from AmWN.

3.2 Argument Mapping Population

After defining the AmWN representation, we next describe our implementation of AmWN. We first populate the AmWN graph with substitution edges for WordNet’s hypernyms and synonyms (as self edges), e.g. ‘buy \Leftrightarrow purchase’ and ‘buy \Rightarrow acquire’. The following subsections describe how mapping edges are created based on various manual and corpus-based information resources.

3.2.1 Nominalization Relations

The relation between a verb and its nominalizations, e.g. between ‘employ’ and ‘employment’, out any current performance improvement.

```

:ORTH "employment"
:VERB "employ"
:VERB-SUBC ((NOM-NP
             :SUBJECT ((DET-POSS)
                       (N-N-MOD)
                       (PP :PVAL ("by"))))
             :OBJECT ((DET-POSS)
                      (PP :PVAL ("of"))))

```

Figure 2: Part of the *employment* Nomlex entry, describing the possible syntactic dependency positions for each role of the transitive frame. It states, for example, that the verbal ‘object’ role can be mapped to *employment* either as a possessive or as the complement of the preposition ‘of’.

is described in WordNet by the *derivationally related* relation. To add argument mappings for these relations we utilize Nomlex-plus (Meyers et al., 2004), a database of around 5000 English nominalizations. Nomlex specifies for each verbal subcategorization frame of each nominalization how its argument positions are mapped to functional roles of related verbs.

For each Nomlex entry, we extract all possible argument mappings between the verbal and nominal forms, as well as between different argument realizations of the noun. For example, the mappings ‘ X_{obj} ’s employment \Leftrightarrow employ X_{obj} ’ and ‘ X_{obj} ’s employment \Leftrightarrow employment of X_{obj} ’ are derived from the entry in Figure 2.

The major challenge in integrating Nomlex and WordNet is to identify for each Nomlex noun which WordNet synsets describe its predicative meanings. For example, one synset of ‘acquisition’ that is derivationally related to ‘acquire’ is not predicative: “*an ability that has been acquired by training*”. We mark noun synsets as predicative if they are (transitive) hyponyms of the *act* high-level synset.

Once predicative synsets are identified, we create, for each synset, a node for each subcategorization frame of its noun members, as found in Nomlex-plus. In some nodes not all original synset members are retained, since not all members share all their frames. Mapping edges are then added between nodes that have the same frame. We add both noun-verb edges and noun self-edges that map different realizations of the same functional role (e.g. ‘ X_{obj} ’s employment \Leftrightarrow employment of X_{obj} ’).

As rich as Nomlex-plus is, it still does not include all nominalizations. For example, the nouns

Lexical Relation	Extracted Mappings
buy \Rightarrow pay	buy for $X \Rightarrow$ pay X X buy \Rightarrow X pay
divorce \Rightarrow marry	divorce from $X \Rightarrow$ marry X divorce from $X \Rightarrow$ X marry
kill \Rightarrow die	kill $X \Rightarrow$ X die kill among $X \Rightarrow$ X die
breathe \Rightarrow inhale	breathe $X \Rightarrow$ inhale X breathe in $X \Rightarrow$ inhale X
remind \Rightarrow remember	remind $X \Rightarrow$ X remember remind of $X \Rightarrow$ remember X
teach \Rightarrow learn	teach $X \Rightarrow$ learn X teach to $X \Rightarrow$ X learn
give \Rightarrow have	give $X \Rightarrow$ have X give to $X \Rightarrow$ X have

Table 2: Some argument mappings for WordNet verb-verb relations discovered by unary-DIRT.

‘divorce’ (related to the verb ‘divorce’) and ‘striking’ are missing. WordNet has a much richer set of nominalizations that we would like to use. To do so, we inherit associated frames and argument realizations for each nominalization synset from its closest hypernym that does appear in Nomlex. Thus, ‘divorce’ inherits its information from ‘separation’ and ‘striking’ inherits from ‘hit’. A by-product of this process is the automatic extension of Nomlex-plus with 5100 new nominalization entries, based on the inherited information⁴.

3.2.2 Verb-Verb Relations

There are two inferential relations between verbs in WordNet that do not preserve argument positions: *cause* and *entailment*. Unlike for nominalizations, there is no broad-coverage manual resource of argument mapping for these relations. Hence, we turn to unsupervised approaches that learn entailment rules from corpus statistics.

Many algorithms were proposed for learning entailment rules between templates from corpora (Lin and Pantel, 2001; Szpektor et al., 2004; Sekine, 2005), but typically with mediocre accuracy. However, we only search for rules between verbs for which WordNet already indicates the existence of an entailment relation and are thus not affected by rules that wrongly relate non-entailing verbs. We acquired a rule-set containing the top 300 rules for every unary template in the Reuters RCV1 corpus⁵ by implementing the unary-DIRT algorithm (Szpektor and Dagan, 2008), which was shown to have relatively high recall compared to other algorithms.

⁴We plan making this extension publicly available as well.

⁵<http://about.reuters.com/researchandstandards/corpus/>

To extract argument mappings, we identify all AmWN node pairs whose synsets are related in WordNet by a *cause* or an *entailment* relation. For each pair, we look for unary-DIRT rules between any pair of members in the entailing and entailed synsets. For example, the synset {buy, purchase} entails {pay}, so we look for rules mapping either ‘buy \Rightarrow pay’ or ‘purchase \Rightarrow pay’. Table 2 presents examples for discovered mappings. While unary-DIRT rules are not annotated with functional roles, they can be derived straightforwardly from the verbal dependency relations available in the rule’s templates. The obtained rules are then added to AmWN as mapping edges.

We only search for rules that map a functional role in the frame of one verb to any role for the other verb. Focusing on frame elements avoids extracting mapping rules learned for adjuncts, which tend to be of low precision.

3.3 Rule Filtering

In preliminary analysis we found two phenomena, *sense drifting* and *rare senses*, which may reduce the effectiveness of AmWN-based inference even if each graph edge by itself, taken out of context, is correct. To address these phenomena within practical inference we propose the following optional methods for rule filtering.

Sense Drifting WordNet verbs typically have a more fine-grained set of synsets than their related nominalizations. There are cases where several verb synsets are related to the same nominal synset. Since entailment between a verb and its nominalization is bidirectional, all such verb synsets would end up entailing each other via the nominal node.

Alas, some of these connected verb synsets represent quite different meanings, which results in incorrect inferences. This problem, which we call *sense drifting*, is demonstrated in Figure 3. To address it, we constrain each rule generation chain to include at most one verb-noun edge, which still connects the noun and verb hierarchies.

Rare Senses Some word senses in WordNet are rare. Thus, applying rules that correspond to such senses yields many incorrect inferences, since they are typically matched against other frequent senses of the word. Such a rule is ‘have $X \Rightarrow X$ is born’, corresponding to a rare sense of ‘have’. WSD is a possible solution for this problem. However, most state-of-the-art IE, QA and RTE sys-

tems do not rely on WSD methods, which are currently not sufficiently robust.

To circumvent the rare sense problem, we instead filter out such rules. Each AmWN rule is validated against our unary-DIRT rule-set, which, being corpus-based, contains mostly rules for frequent senses. A rule is *directly-validated* if it is in the corpus-based rule-set, or if it is a nominal-verb rule which describes a reliable morphological change for a predicate. The AmWN graph-path that generated each rule is automatically examined. A rule is considered *valid* if there is a sequence of directly-validated intermediate rules along the path whose transitive chaining generates the rule. Invalid rules are filtered out.

To illustrate, suppose the rule ‘a \Rightarrow d’ was generated by the chain ‘a \Rightarrow b \Rightarrow c \Rightarrow d’. It is valid if there is a rule chain along the path that yields ‘a \Rightarrow d’, e.g. {‘a \Rightarrow b’, ‘b \Rightarrow c’, ‘c \Rightarrow d’} or {‘a \Rightarrow b’, ‘b \Rightarrow d’}, whose rules are all directly-validated.

4 Experimental Setup

We follow here the experimental setup presented in (Szpektor and Dagan, 2008), testing the generated rules on the ACE 2005 event dataset⁶. This standard IE benchmark includes 33 types of event predicates such as *Injure*, *Sue* and *Divorce*⁷. The ACE guidelines specify for each event its possible arguments. For example, some of the *Injure* event arguments are *Agent* and *Victim*. All event mentions, including their instantiated arguments, are annotated in a corpus collected from various sources (newswire articles, blogs, etc.).

To utilize the ACE dataset for evaluating rule applications, each ACE event predicate was represented by a set of unary *seed templates*, one for each event argument. Example seed templates for *Injure* are ‘A injure’ and ‘injure V’. Each event argument is mapped to the corresponding seed template variable, e.g. ‘Agent’ to A and ‘Victim’ to V in the above example.

We manually annotated each seed template with a subcategorization frame and an argument functional role, e.g. ‘injure_{trans} V_{obj}’. We also included relevant WordNet synset-ids, so only rules fitting the target meaning of the event will be extracted. In this experiment, we focused only on the core semantic arguments. Adjuncts (time and

⁶<http://projects ldc.upenn.edu/ace/>

⁷Only 26 frequent event types that correspond to a unique predicate were tested, following (Szpektor and Dagan, 2008).

	Synset Members	WordNet Gloss
	(verb) collar, nail, apprehend, arrest , pick up, nab, cop	take into custody
	↕	
	(noun) apprehension, arrest, catch, collar, pinch, taking into custody	the act of apprehending (especially apprehending a criminal)
	↕	
	(verb) get, catch, capture	succeed in catching or seizing, especially after a chase
	↕	
	(noun) capture, seizure	the act of taking of a person by force
	↕	
	(verb) seize	take or capture by force
	↑ (hyponym)	
	(verb) kidnap , noble, abduct, snatch	take away to an undisclosed location against their will and usually in order to extract a ransom

Figure 3: A WordNet sense-drifting traversal, generating the incorrect inference ‘kidnap \Rightarrow arrest’.

place) were ignored since they typically don’t require argument mapping, the main target for our assessment.

The ACE corpus was dependency-parsed with Minipar (Lin, 1998) and annotated with functional roles and frames for each predicate mention. The functional roles for a verb mention were taken directly from the corresponding dependency tree relations. Its frame was chosen to be the largest WordNet frame of that verb that matched the mention’s roles.

Nominalization frames and functional roles in the text were annotated using our extended Nomlex-plus database. For each nominal mention, we found the largest Nomlex frame whose syntactic argument positions matched those of the mention’s arguments. The arguments were then annotated with the specified roles of the chosen frame. Ambiguous cases, where the same argument position could match multiple roles, were left unannotated, as discussed in Section 2.

Argument mentions for events were found in the annotated corpus by matching either the seed templates or the templates entailing them in some rules. The matching procedure follows the one described in Section 2. Templates are matched using a syntactic matcher that handles simple syntactic variations such as passive-form and conjunctions. For example, ‘wound_{trans} V_{obj} \Rightarrow injure_{trans} V_{obj}’ was matched in the text “Hagel_{obj} was wounded_{trans} in Vietnam”. A rule application is considered correct if the matched argument is annotated in the corpus with the corresponding ACE role.

We note that our system performance on the ACE task as such is limited. First, WordNet does not provide all types of needed rules. Second, the system of our experimental setting is rather basic,

with limited matching capabilities and without a WSD module. However, this test-set is still very useful for relative comparison of WordNet and our proposed AmWN.

5 Results and Analysis

We tested four different rule-set configurations: a) only the seed templates, without any rules; b) rules generated based on WordNet 3.0 without argument mapping, using only synonym and hypernym relations; c) WordNet rules from (b), filtered using our corpus-based validation method for rare senses; d) rules generated from our AmWN.

Out of the 8953 non-substitutable inferential relations that we identified in WordNet, our AmWN implementation created mapping edges for 75% of 8325 Noun-Verb relations and 70% of 628 Verb-Verb relations. Altogether 41549 mapping edges between synset nodes were added. A manual error analysis of these mappings is provided in Section 5.2.

Each configuration was evaluated for each ACE event. We measured the percentage of correct argument mentions extracted out of all correct argument mentions annotated for the event (recall) and out of all argument mentions extracted (precision), and F1, their harmonic average. We report macro averages over the 26 event types.

5.1 Results

Table 3 summarizes the results for the different configurations. As expected, matching only the seed templates yields the highest precision but lowest recall. Using the standard WordNet configuration actually decreases overall F1 performance. Though recall increases relatively by 30%, thanks to WordNet expansions, F1 is penalized by a sharp

Configuration	R (%)	P (%)	F1
No Rules	13.5	63.0	20.7
WordNet	17.5	35.3	18.5
WordNet with rule validation	16.5	46.9	20.4
AmWN	20.8	43.9	24.2

Table 3: Recall (R), Precision (P) and F1 results for the different tested configurations.

relative drop in precision (by 56%). The main reason for this decline is the application of rules involving infrequent word senses, as elaborated in Section 3.3.

When our rule validation approach is applied to standard WordNet expansions, a much higher precision is achieved with only a small decline in recall. This shows that our corpus-based filtering method manages to avoid many of the noisy rules for rare senses, while maintaining those that are frequently involved in inference.

Finally, our main result shows that adding argument mapping improves performance substantially. AmWN achieves a much higher recall than WordNet. Recall increases relatively by 26% over validated WordNet, and by 54% over the no-rules baseline. Furthermore, precision drops only slightly, by 6%, compared to validated WordNet. This shows that argument mapping increases WordNet’s graph connectivity, while our rule-validation method maintains almost the same precision for many more generated rules. The improvement in overall F1 performance is statistically significant compared to all other configurations, according to the two-sided Wilcoxon signed rank test at the level of 0.01 (Wilcoxon, 1945).

5.2 Error Analysis

We manually analyzed the reasons for false positives (incorrect extractions) and false negatives (missed extractions) of AmWN by sampling 300 extractions of each type.

From the false positives analysis (Table 4) we see that practically all generated rules are correct (99.4%), that is, they would be valid in some contexts. Almost all errors come from matching errors (including parse errors) and context mismatches, due to our limited IE implementation. The only two incorrect rules sampled were due to an incorrect Nomlex entry and a WordNet synset that should have been split into two separate senses. Considering that correct extractions resulted, per our analysis, from correct rules, the analysis of this

Reason	% mentions
Context mismatch	57.2
Match error	33.6
Errors in gold-standard annotation	8.6
Incorrect Rule learned	0.6

Table 4: Distribution of reasons for false positives (incorrect argument extractions).

Reason	% mentions
Rule not learned	67.7
Match error	18.0
Discourse analysis needed	12.0
Argument is predicative	1.3
Errors in gold-standard annotation	1.0

Table 5: Distribution of reasons for false negatives (missed argument mentions).

sample indicates that virtually all AmWN edges that get utilized in practice are correct.

Context mismatches, which constitute the majority of errors (57.2%), occur when the entailing template of a rule is matched in inappropriate contexts. This occurs typically when the match is against another sense of the predicate, or when an argument is not of the requested type (e.g. “*The Enron sentence*” vs. “*A one month sentence*”). In future work, we plan to address this problem by utilizing context-sensitive application of rules in the spirit of (Szpektor et al., 2008).

Table 5 presents the false negatives analysis. Most missed extractions are due to rules that were not learned (67.7%). These mainly involve complex templates (‘file a lawsuit \Leftrightarrow sue’) and inference rules that are not synonyms/hypernyms (‘execute \Rightarrow sentence’), which are not widely annotated in WordNet. From further analysis, we found that 10% of these misses are due to rules that are generated from AmWN but filtered out by one of our filtering methods (Section 3.3).

12% of the arguments cannot be extracted by rules alone, due to required discourse analysis, while 18% of the mentions were missed due to incorrect syntactic matching. By assuming correct matches in these cases and avoiding rule filtering, we can estimate the upper bound recall of the rule-set for the ACE dataset to be 40%.

In conclusion, for better performance the system should be augmented with context modeling and better template matching. Additionally, other rule-bases, e.g. DIRT (Lin and Pantel, 2001), should be added to increase rule coverage.

Configuration	R (%)	P (%)	F1
AmWN	20.8	43.9	24.2
No nominalization mappings	18.1	45.5	21.8
No verb-verb mappings	19.3	43.8	22.8
No rule validation	22.0	30.4	20.9
No sense drift blocking	22.5	37.4	21.7

Table 6: The Recall (R), Precision (P) and F1 results for ablation tests.

5.3 Component Analysis

Table 6 presents ablation tests that assess the marginal contribution of each AmWN component. Nominal-verb and verb-verb mappings contribute to the graph connectivity, hence the recall reduction when they are removed.

Complementary to recall components, rule filtering improves precision. When removing the corpus-based rule-validation, recall increases relatively by 6% but precision drops relatively by 30%, showing the benefit of noisy-rule filtering. Allowing sense drifting hurts precision, a relative drop of 22%. Yet, recall increases relatively by 8%, indicating that some verb synsets, connected via a shared nominal, entail each other even though they are not connected directly. For example, ‘found $X \Leftrightarrow$ create X ’ was generated only via the shared nominal ‘founding’. In future work, we plan to apply AmWN to a coarse-grained set of WordNet synsets (Palmer et al., 2007) as a possible solution to sense drifting.

6 Related Work

Several works attempt to extend WordNet with additional lexical semantic information (Moldovan and Rus, 2001; Snow et al., 2006; Suchanek et al., 2007; Clark et al., 2008). However, the only previous work we are aware of that enriches WordNet with argument mappings is (Novischi and Moldovan, 2006). This work utilizes VerbNet’s subcategorization frames to identify possible verb arguments. Argument mapping is provided only between verbs, ignoring relations between verbs and nouns. Arguments are mapped based on thematic role names shared between frames of different verbs. However, the semantic interpretation of thematic roles is generally inconsistent across verbs (Lowe et al., 1997; Kaisser and Webber, 2007). Instead, we discover these mappings from corpus statistics, offering an accurate approach (as analyzed in Section 5.2).

A frame semantics approach for argument

mapping between predicates is proposed by the FrameNet project (Baker et al., 1998). Currently, FrameNet is the only resource for frame-semantic argument mappings. However, it is manually constructed and currently covers much less predicates and relations than WordNet. Furthermore, frame-semantic parsers are less robust than syntactic parsers, presently hindering the utilization of this approach in applications (Burchardt and Penacchiotti, 2008).

Nomlex argument mapping patterns similar to ours were derived for IE in (Meyers et al., 1998), but they were not integrated with any additional information, such as WordNet.

7 Conclusions

We presented Argument-mapped WordNet (AmWN), a novel framework for augmenting WordNet with argument mappings at the syntactic representation level. With AmWN, non-substitutable WordNet relations can also be utilized correctly, increasing the coverage of WordNet-based inference. The standard entailment rule representation is augmented in our work with functional roles and subcategorization frames, shown to be a feasible extension needed for correct rule application in general.

Our implementation of AmWN populates WordNet with mappings based on combining manual and corpus-based resources. It covers a broader range of relations compared to prior work and yields more accurate mappings. We also introduced a novel corpus-based validation mechanism, avoiding rules for infrequent senses. Our experiments show that AmWN substantially improves standard WordNet-based inference.

In future work we plan to add mappings between verbs and adjectives and between different frames of a verb. We also want to incorporate resources for additional subcategorization frames, such as VerbNet. Finally, we plan to enhance our text annotation based on noun-compound disambiguation (Lapata and Lascarides, 2003).

Acknowledgements

This work was partially supported by the NEGEV project (www.negev-initiative.org), the PASCAL-2 Network of Excellence of the European Community FP7-ICT-2007-1-216886, the FBK-irst/Bar-Ilan University collaboration and the Israel Science Foundation grant 1112/08.

References

- Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of ACL*.
- Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge. In *Second PASCAL Challenge Workshop for Recognizing Textual Entailment*.
- Regina Barzilay and Michael Elhadad. 1997. Using lexical chains for text summarization. In *Proceedings of ACL*.
- Aljoscha Burchardt and Marco Pennacchiotti. 2008. Fate: a framenet-annotated corpus for textual entailment. In *Proceedings of LREC*.
- Peter Clark, Christiane Fellbaum, Jerry R. Hobbs, Phil Harrison, William R. Murray, and John Thompson. 2008. Augmenting WordNet for Deep Understanding of Text. In *Proceedings of STEP 2008*.
- Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*.
- Michael Kaisser and Bonnie Webber. 2007. Question answering based on semantic roles. In *ACL 2007 Workshop on Deep Linguistic Processing*.
- Karin Kipper, Hoa Trang Dang, and Martha Palmer. 2000. Class-based construction of a verb lexicon. In *Proceedings of AAAI*.
- Mirella Lapata and Alex Lascarides. 2003. Detecting novel compounds: The role of distributional evidence. In *Proceedings of EACL*.
- Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360.
- Dekang Lin. 1998. Dependency-based evaluation of minipar. In *Proceedings of the Workshop on Evaluation of Parsing Systems at LREC 1998*, Granada, Spain.
- John B. Lowe, Collin F. Baker, and Charles J. Fillmore. 1997. A frame-semantic approach to semantic annotation. In *Proceedings of the SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*
- Catherine Macleod, Ralph Grishman, Adam Meyers, Leslie Barrett, and Ruth Reeves. 1998. NOMLEX: A Lexicon of Nominalizations. In *Proceedings of EURALEX*.
- Adams Meyers, Catherine Macleod, Roman Yangarber, Ralph Grishman, Leslie Barrett, and Ruth Reeves. 1998. Using nomlex to produce nominalization patterns for information extraction. In *Proceedings of COLING*.
- Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekeley, Veronkia Zielinska, and Brian Young. 2004. The Cross-Breeding of Dictionaries. In *Proceedings of LREC*.
- George A. Miller. 1995. Wordnet: A lexical database for english. In *Communications of the ACM*.
- Dan Moldovan and Rada Mihalcea. 2000. Using wordnet and lexical operators to improve internet searches. *IEEE Internet Computing*, 4(1):34–43.
- Dan Moldovan and Adrian Novischi. 2002. Lexical chains for question answering. In *Proceedings of COLING*.
- Dan Moldovan and Vasile Rus. 2001. Logic form transformation of wordnet and its applicability to question answering. In *Proceedings of ACL*.
- Adrian Novischi and Dan Moldovan. 2006. Question answering with lexical chains propagating verb arguments. In *Proceedings of ACL*.
- 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.
- Marius Pasca and Sanda Harabagiu. 2001. The informative role of wordnet in open-domain question answering. In *Proceedings of Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*.
- Satoshi Sekine. 2005. Automatic paraphrase discovery based on context and keywords between ne pairs. In *Proceedings of IWP*.
- Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of ACL*.
- Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge - unifying wordnet and wikipedia. In *Proceedings of WWW2007*.
- Idan Szpektor and Ido Dagan. 2008. Learning entailment rules for unary templates. In *Proceedings of COLING*.
- Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventure Coppola. 2004. Scaling web based acquisition of entailment patterns. In *Proceedings of EMNLP 2004*.
- Idan Szpektor, Ido Dagan, Roy Bar-Haim, and Jacob Goldberger. 2008. Contextual preferences. In *Proceedings of ACL*.
- Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1:80–83.