A Robust Approach to Aligning Heterogeneous Lexical Resources

Mohammad Taher Pilehvar and **Roberto Navigli**
Department of Computer Science
Sapienza University of Rome
{pilehvar,navigli}@di.uniroma1.it

Abstract

Lexical resource alignment has been an active field of research over the last decade. However, prior methods for aligning lexical resources have been either specific to a particular pair of resources, or heavily dependent on the availability of hand-crafted alignment data for the pair of resources to be aligned. Here we present a unified approach that can be applied to an arbitrary pair of lexical resources, including machine-readable dictionaries with no network structure. Our approach leverages a similarity measure that enables the structural comparison of senses across lexical resources, achieving state-of-the-art performance on the task of aligning WordNet to three different collaborative resources: Wikipedia, Wiktionary and OmegaWiki.

1 Introduction

Lexical resources are repositories of machine-readable knowledge that can be used in virtually any Natural Language Processing task. Notable examples are WordNet, Wikipedia and, more recently, collaboratively-curated resources such as OmegaWiki and Wiktionary (Hovy et al., 2013). On the one hand, these resources are heterogeneous in design, structure and content, but, on the other hand, they often provide complementary knowledge which we would like to see integrated. Given the large scale this intrinsic issue can only be addressed automatically, by means of lexical resource alignment algorithms. Owing to its ability to bring together features like multilinguality and increasing coverage, over the past few years resource alignment has proven beneficial to a wide spectrum of tasks, such as Semantic Parsing (Shi and Mihalcea, 2005), Semantic Role Labeling (Palmer et al., 2010), and Word Sense Disambiguation (Navigli and Ponzetto, 2012).

Nevertheless, when it comes to aligning textual definitions in different resources, the lexical approach (Ruiz-Casado et al., 2005; de Melo and Weikum, 2010; Henrich et al., 2011) falls short because of the potential use of totally different wordings to define the same concept. Deeper approaches leverage semantic similarity to go beyond the surface realization of definitions (Navigli, 2006; Meyer and Gurevych, 2011; Niemann and Gurevych, 2011). While providing good results in general, these approaches fail when the definitions of a given word are not of adequate quality and expressiveness to be distinguishable from one another. When a lexical resource can be viewed as a semantic graph, as with WordNet or Wikipedia, this limit can be overcome by means of alignment algorithms that exploit the network structure to determine the similarity of concept pairs. However, not all lexical resources provide explicit semantic relations between concepts and, hence, machine-readable dictionaries like Wiktionary have first to be transformed into semantic graphs before such graph-based approaches can be applied to them. To do this, recent work has proposed graph construction by monosemous linking, where a concept is linked to all the concepts associated with the monosemous words in its definition (Matuschek and Gurevych, 2013). However, this alignment method still involves tuning of parameters which are highly dependent on the characteristics of the generated graphs and, hence, requires hand-crafted sense alignments for the specific pair of resources to be aligned, a task which has to be replicated every time the resources are updated.

In this paper we propose a unified approach to aligning arbitrary pairs of lexical resources which is independent of their specific structure. Thanks to a novel modeling of the sense entries and an effective ontologization algorithm, our approach also fares well when resources lack relational structure or pair-specific training data is absent, meaning that it is applicable to arbitrary pairs without adaptation. We report state-of-the-art performance when aligning WordNet to Wikipedia, OmegaWiki and Wiktionary.

2 Resource Alignment

Preliminaries. Our approach for aligning lexical resources exploits the graph structure of each resource. Therefore, we assume that a lexical resource L can be represented as an undirected graph $G = (V, E)$ where V is the set of nodes, i.e., the concepts defined in the resource, and E is the set of undirected edges, i.e., semantic relations between concepts. Each concept $c \in V$ is associated with a set of lexicalizations $\mathcal{L}_G(c) = \{w_1, w_2, ..., w_n\}$. For instance, WordNet can be readily represented as an undirected graph G whose nodes are synsets and edges are modeled after the relations between synsets defined in WordNet (e.g., hypernymy, meronymy, etc.), and $\mathcal{L}_G$ is the mapping between each synset node and the set of synonyms which express the concept. However, other resources such as Wiktionary do not provide semantic relations between concepts and, therefore, have first to be transformed into semantic networks before they can be aligned using our alignment algorithm. We explain in Section 3 how a semi-structured resource which does not exhibit a graph structure can be transformed into a semantic network.

Alignment algorithm. Given a pair of lexical resources L_1 and L_2, we align each concept in L_1 by mapping it to its corresponding concept(s) in the target lexicon L_2. Algorithm 1 formalizes the alignment process: the algorithm takes as input the semantic graphs G_1 and G_2 corresponding to the two resources, as explained above, and produces as output an alignment in the form of a set A of concept pairs. The algorithm iterates over all concepts $c_1 \in V_1$ and, for each of them, obtains the set of concepts $C \subset V_2$, which can be considered as alignment candidates for c_1 (line 3). For a concept c_1, alignment candidates in G_2 usually consist of every concept $c_2 \in V_2$ that shares at least one lexicalization with c_1 in the same part of speech tag, i.e., $\mathcal{L}_{G_1}(c_1) \cap \mathcal{L}_{G_2}(c_2) \neq \emptyset$ (Reiter et al., 2008; Meyer and Gurevych, 2011). Once the set of target candidates C for a source concept c_1 is obtained, the alignment task can be cast as that of identifying those concepts in C to which c_1 should be aligned. To do this, the algorithm calculates the similarity between c_1 and each $c_2 \in C$ (line 5). If their similarity score exceeds a certain value denoted by θ (line 6), the two concepts c_1 and c_2 are aligned and the pair (c_1, c_2) is added to A (line 7).

Algorithm 1 Lexical Resource Aligner

```
Input: graphs $H = (V_H, E_H)$, $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, the similarity threshold $\theta$, and the combination parameter $\beta$
Output: $A$, the set of all aligned concept pairs
1: $A \leftarrow \emptyset$
2: for each concept $c_1 \in V_1$
3:     $C \leftarrow$ getCandidates$(c_1, V_2)$
4:     for each concept $c_2 \in C$
5:         $sim \leftarrow$ calculateSimilarity$(H, G_1, G_2, c_1, c_2, \beta)$
6:         if $sim > \theta$ then
7:             $A \leftarrow A \cup \{(c_1, c_2)\}$
8: return $A$
```

Different resource alignment techniques usually vary in the way they compute the similarity of a pair of concepts across two resources (line 5 in Algorithm 1). In the following, we present our novel approach for measuring the similarity of concept pairs.

2.1 Measuring the Similarity of Concepts

Figure 1 illustrates the procedure underlying our cross-resource concept similarity measurement technique. As can be seen, the approach consists of two main components: *definitional similarity* and *structural similarity*. Each of these components gets, as its input, a pair of concepts belonging to two different semantic networks and produces a similarity score. These two scores are then combined into an overall score (part (e) of Figure 1) which quantifies the semantic similarity of the two input concepts c_1 and c_2.

The definitional similarity component computes the similarity of two concepts in terms of the similarity of their definitions, a method that has also been used in previous work for aligning lexical resources (Niemann and Gurevych, 2011; Henrich et al., 2012). In spite of its simplicity, the mere calculation of the similarity of concept definitions provides a strong baseline, especially for cases where the definitional texts for a pair of concepts to be aligned are lexically similar, yet distinguishable from the other definitions. However, as mentioned in the introduction, definition similarity-based techniques fail at identifying the correct alignments in cases where different wordings are used or definitions are not of high quality. The structural similarity component, instead, is a novel graph-based similarity measurement technique which calculates the similarity between a pair of concepts across the semantic networks of the two resources by leveraging the semantic

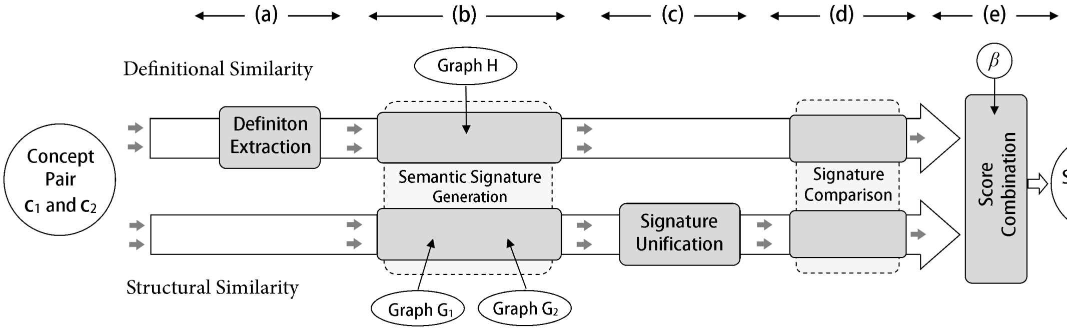

Figure 1: The process of measuring the similarity of a pair of concepts across two resources. The method consists of two components: definitional and structural similarities, each measuring a similarity score for the given concept pair. The two scores are combined by means of parameter β in the last stage.

structure of those networks. This component goes beyond the surface realization of concepts, thus providing a deeper measure of concept similarity.

The two components share the same backbone (parts (b) and (d) of Figure 1), but differ in some stages (parts (a) and (c) in Figure 1). In the following, we explain all the stages involved in the two components (gray blocks in the figure).

2.1.1 Semantic signature generation

The aim of this stage is to model a given concept or set of concepts through a vectorial semantic representation, which we refer to as the **semantic signature** of the input. We utilized Personalized PageRank (Haveliwala, 2002, PPR), a random walk graph algorithm, for calculating semantic signatures. The original PageRank (PR) algorithm (Brin and Page, 1998) computes, for a given graph, a single vector wherein each node is associated with a weight denoting its structural importance in that graph. PPR is a variation of PR where the computation is biased towards a set of initial nodes in order to capture the notion of importance with respect to those particular nodes. PPR has been previously used in a wide variety of tasks such as definition similarity-based resource alignment (Niemann and Gurevych, 2011), textual semantic similarity (Hughes and Ramage, 2007; Pilehvar et al., 2013), Word Sense Disambiguation (Agirre and Soroa, 2009; Faralli and Navigli, 2012) and semantic text categorization (Navigli et al., 2011). When applied to a semantic graph by initializing the random walks from a set of concepts (nodes), PPR yields a vector in which each concept is associated with a weight denoting its semantic relevance to the initial concepts.

Formally, we first represent a semantic network consisting of N concepts as a row-stochastic transition matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$. The cell (i, j) in the matrix denotes the probability of moving from a concept i to j in the graph: 0 if no edge exists from i to j and $1/degree(i)$ otherwise. Then the PPR vector, hence the semantic signature $\mathcal{S}_\mathbf{v}$ of vector $\mathbf{v}$ is the unique solution to the linear system: $\mathcal{S}_\mathbf{v} = (1 - \alpha)\,\mathbf{v} + \alpha\,\mathbf{M}\,\mathcal{S}_\mathbf{v}$, where $\mathbf{v}$ is the personalization vector of size N in which all the probability mass is put on the concepts for which a semantic signature is to be computed and α is the damping factor, which is usually set to 0.85 (Brin and Page, 1998). We used the UKB[1] off-the-shelf implementation of PPR.

Definitional similarity signature. In the definitional similarity component, the two concepts c_1 and c_2 are first represented by their corresponding definitions d_1 and d_2 in the respective resources L_1 and L_2 (Figure 1(a), top). To improve expressiveness, we follow Niemann and Gurevych (2011) and further extend d_i with all the word forms associated with concept c_i and its neighbours, i.e., the union of all lexicalizations $\mathcal{L}_{G_i}(x)$ for all concepts $x \in \{c' \in V_i : (c, c') \in E_i\} \cup \{c\}$, where E_i is the set of edges in G_i. In this component the personalization vector $\mathbf{v}_i$ is set by uniformly distributing the probability mass over the nodes corresponding to the senses of all the content words in the extended definition of d_i according to the sense inventory of a semantic network H. We use the same semantic graph H for computing the semantic signatures of both definitions. Any semantic network with a dense relational structure, providing good coverage of the words appearing in the definitions, is a suitable candidate for H. For this purpose we used the WordNet (Fellbaum, 1998) graph which was further enriched by connecting

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

each concept to all the concepts appearing in its disambiguated gloss.[2]

Structural similarity signature. In the structural similarity component (Figure 1(b), bottom), the semantic signature for each concept c_i is computed by running the PPR algorithm on its corresponding graph G_i, hence a different $\mathbf{M}_i$ is built for each of the two concepts.

2.1.2 Signature unification

As mentioned earlier, semantic signatures are vectors with dimension equal to the number of nodes in the semantic graph. Since the structural similarity signatures $\mathcal{S}_{\mathbf{v}_1}$ and $\mathcal{S}_{\mathbf{v}_2}$ are calculated on different graphs and thus have different dimensions, we need to make them comparable by unifying them. We therefore propose an approach (part (c) of Figure 1) that finds a common ground between the two signatures: to this end we consider all the concepts associated with monosemous words in the two signatures as landmarks and restrict the two signatures exclusively to those common concepts. Leveraging monosemous words as bridges between two signatures is a particularly reliable technique as typically a significant portion of all words in a lexicon are monosemous.[3]

Formally, let $\mathcal{I}_G(w)$ be an inventory mapping function that maps a term w to the set of concepts which are expressed by w in graph G. Then, given two signatures $\mathcal{S}_{\mathbf{v}_1}$ and $\mathcal{S}_{\mathbf{v}_2}$, computed on the respective graphs G_1 and G_2, we first obtain the set $\mathcal{M}$ of words that are monosemous according to both semantic networks, i.e., $\mathcal{M} = \{w : |\mathcal{I}_{G_1}(w)| = 1 \wedge |\mathcal{I}_{G_2}(w)| = 1\}$. We then transform each of the two signatures $\mathcal{S}_{\mathbf{v}_i}$ into a new sub-signature $\mathcal{S}'_{\mathbf{v}_i}$ whose dimension is $|\mathcal{M}|$: the k^{th} component of $\mathcal{S}'_{\mathbf{v}_i}$ corresponds to the weight in $\mathcal{S}_{\mathbf{v}_i}$ of the only concept of w_k in $\mathcal{I}_{G_i}(w_k)$. As an example, assume we are given two semantic signatures computed for two concepts in WordNet and Wiktionary. Also, consider the noun *tradeoff* which is monosemous according to both these resources. Then, each of the two unified sub-signatures will contain a component whose weight is determined by the weight of the only concept associated with *tradeoff*$_n$ in the corresponding semantic signature. As a result of the unification process, we obtain a pair of equally-sized semantic signatures with comparable components.

2.1.3 Signature comparison

Having at hand the semantic signatures for the two input concepts, we proceed to comparing them (part (d) in Figure 1). We leverage a non-parametric measure proposed by Pilehvar et al. (2013) which first transforms each signature into a list of sorted elements and then calculates the similarity on the basis of the average ranking of elements across the two lists:

$$Sim(\mathcal{S}_{\mathbf{v}_1}, \mathcal{S}_{\mathbf{v}_2}) = \frac{\sum_{i=1}^{|T|} (r_i^1 + r_i^2)^{-1}}{\sum_{i=1}^{|T|} (2i)^{-1}} \qquad (1)$$

where T is the intersection of all concepts with non-zero probability in the two signatures and r_i^j is the rank of the i^{th} entry in the j^{th} sorted list. The denominator is a normalization factor to guarantee a maximum value of one. The method penalizes the differences in the higher rankings more than it does for the lower ones. The measure was shown to outperform the conventional cosine distance when comparing different semantic signatures in multiple textual similarity tasks (Pilehvar et al., 2013).

2.1.4 Score combination

Finally (part (e) of Figure 1), we calculate the overall similarity between two concepts as a linear combination of their definitional and structural similarities: $\beta\, Sim_{def}(\mathcal{S}_{\mathbf{v}_1}, \mathcal{S}_{\mathbf{v}_2}) + (1 - \beta)\, Sim_{str}(\mathcal{S}_{\mathbf{v}_1}, \mathcal{S}_{\mathbf{v}_2})$. In Section 4.2.1, we explain how we set, in our experiments, the values of β and the similarity threshold θ (cf. alignment algorithm in Section 2).

3 Lexical Resource Ontologization

In Section 2, we presented our approach for aligning lexical resources. However, the approach assumes that the input resources can be viewed as semantic networks, which seems to limit its applicability to structured resources only. In order to address this issue and hence generalize our alignment approach to any given lexical resource, we propose a method for transforming a given machine-readable dictionary into a semantic network, a process we refer to as *ontologization*.

Our ontologization algorithm takes as input a lexicon L and outputs a semantic graph $G = (V, E)$ where, as already defined in Section 2, V is the set of concepts in L and E is the set of semantic relations between these concepts. Introducing relational links into a lexicon can be achieved in different ways. A first option is to extract binary

[2] http://wordnet.princeton.edu

[3] For instance, we calculated that more than 80% of the words in WordNet are monosemous, with over 60% of all the synsets containing at least one of them.

relations between pairs of words from raw text. Both words in these relations, however, should be disambiguated according to the given lexicon (Pantel and Pennacchiotti, 2008), making the task particularly prone to mistakes due to the high number of possible sense pairings.

Here, we take an alternative approach which requires disambiguation on the target side only, hence reducing the size of the search space significantly. We first create the empty undirected graph $G_L = (V, E)$ such that V is the set of concepts in L and $E = \emptyset$. For each source concept $c \in V$ we create a bag of content words $W = \{w_1, \ldots, w_n\}$ which includes all the content words in its definition d and, if available, additional related words obtained from lexicon relations (e.g., synonyms in Wiktionary). The problem is then cast as a disambiguation task whose goal is to identify the intended sense of each word $w_i \in W$ according to the sense inventory of L: if w_i is monosemous, i.e., $|\{\mathcal{I}_{G_L}(w_i)\}| = 1$, we connect our source concept c to the only sense c_{w_i} of w_i and set $E := E \cup \{\{c, c_{w_i}\}\}$; else, w_i has multiple senses in L. In this latter case, we choose the most appropriate concept $c_i \in \mathcal{I}_{G_L}(w_i)$ by finding the maximal similarity between the definition of c and the definitions of each sense of w_i. To do this, we apply our definitional similarity measure introduced in Section 2.1. Having found the intended sense $\hat{c}_{w_i}$ of w_i, we add the edge $\{c, \hat{c}_{w_i}\}$ to E. As a result of this procedure, we obtain a semantic graph representation G for the lexicon L.

As an example, consider the 4^{th} sense of the noun *cone* in Wiktionary (i.e., *cone*^4_n) which is defined as *"The fruit of a conifer"*. The definition contains two content words: *fruit*$_n$ and *conifer*$_n$. The latter word is monosemous in Wiktionary, hence we directly connect *cone*^4_n to the only sense of *conifer*$_n$. The noun *fruit*, however, has 5 senses in Wiktionary. We therefore measure the similarity between the definition of *cone*^4_n and all the 5 definitions of *fruit* and introduce a link from *cone*^4_n to the sense of fruit which yields the maximal similarity value (defined as *"(botany) The seed-bearing part of a plant..."*).

4 Experiments

Lexical resources. To enable a comparison with the state of the art, we followed Matuschek and Gurevych (2013) and performed an alignment of WordNet synsets (WN) to three different collaboratively-constructed resources: Wikipedia (WP), Wiktionary (WT), and OmegaWiki (OW). We utilized the DKPro software (Zesch et al., 2008; Gurevych et al., 2012) to access the information in the foregoing three resources. For WP, WT, OW we used the dump versions 20090822, 20131002, and 20131115, respectively.

Evaluation measures. We followed previous work (Navigli and Ponzetto, 2012; Matuschek and Gurevych, 2013) and evaluated the alignment performance in terms of four measures: precision, recall, F1, and accuracy. Precision is the fraction of correct alignment judgments returned by the system and recall is the fraction of alignment judgments in the gold standard dataset that are correctly returned by the system. F1 is the harmonic mean of precision and recall. We also report results for accuracy which, in addition to true positives, takes into account true negatives, i.e., pairs which are correctly judged as unaligned.

Lexicons and semantic graphs. Here, we describe how the four semantic graphs for our four lexical resources (i.e., WN, WP, WT, OW) were constructed. As mentioned in Section 2.1.1, we build the WN graph by including all the synsets and semantic relations defined in WordNet (e.g., hypernymy and meronymy) and further populate the relation set by connecting a synset to all the other synsets that appear in its disambiguated gloss. For WP, we used the graph provided by Matuschek and Gurevych (2013), constructed by directly connecting an article (concept) to all the hyperlinks in its first paragraph, together with the category links. Our WN and WP graphs have 118K and 2.8M nodes, respectively, with the average node degree being roughly 9 in both resources.

The other two resources, i.e., WT and OW, do not provide a reliable network of semantic relations, therefore we used our ontologization approach to construct their corresponding semantic graphs. We report, in the following subsection, the experiments carried out to assess the accuracy of our ontologization method, together with the statistics of the obtained graphs for WT and OW.

4.1 Ontologization Experiments

For ontologizing WT and OW, the bag of content words W is given by the content words in sense definitions and, if available, additional related words obtained from lexicon relations (see Section 3). In WT, both of these are in word surface form and hence had to be disambiguated. For OW, however, the encoded relations, though rela-

Source	Type	WT	OW
Definition	Ambiguous	76.6%	50.7%
	Unambiguous	18.3%	32.9%
Relation	Ambiguous	2.8%	-
	Unambiguous	2.3%	16.4%
Total number of edges		2.1M	255K

Table 1: The statistics of the generated graphs for WT and OW. We report the distribution of the edges across types (i.e., ambiguous and unambiguous) and sources (i.e., definitions and relations) from which candidate words were obtained.

Approach	P	R	F1	A
WKTWSD	0.780	**0.800**	0.790	0.840
Our method	**0.852**	0.767	**0.807**	**0.857**
Human	-	-	0.890	0.910

Table 2: The performance of relation disambiguation for our similarity-based disambiguation method, as well as for the WKTWSD system.

tively small in number, are already disambiguated and, therefore, the ontologization was just performed on the definition's content words.

The resulting graphs for WT and OW contain 430K and 48K nodes, respectively, each providing more than 95% coverage of concepts, with the average node degree being around 10 for both resources. We present in Table 1, for WT and OW, the total number of edges together with their distribution across types (i.e., ambiguous and unambiguous) and sources (i.e., definitions and relations) from which candidate words were obtained.

The edges obtained from unambiguous entries are essentially sense disambiguated on both sides whereas those obtained from ambiguous terms are a result of our similarity-based disambiguation. Hence, given that a large portion of edges came from ambiguous words (see Table 1), we carried out an experiment to evaluate the accuracy of our disambiguation method. To this end, we took as our benchmark the dataset provided by Meyer and Gurevych (2010) for evaluating relation disambiguation in WT. The dataset contains 394 manually-disambiguated relations. We compared our similarity-based disambiguation approach against the state of the art on this dataset, i.e., the WKTWSD system, which is a WT relation disambiguation algorithm based on a series of rules (Meyer and Gurevych, 2012b).

Table 2 shows the performance of our disambiguation method, together with that of WKTWSD, in terms of Precision (P), Recall (R), F1, and accuracy. The "Human" row corresponds to the inter-rater F1 and accuracy scores, i.e., the upper-bound performance on this dataset, as calculated by Meyer and Gurevych (2010). As can be seen, our method proves to be very accurate, surpassing the performance of the WKTWSD system in terms of precision, F1, and accuracy. This is particularly interesting as the WKTWSD system uses a rule-based technique specific to relation disambiguation in WT, whereas our method is resource independent and can be applied to arbitrary words in the definition of any concept. We also note that the graph constructed by Meyer and Gurevych (2010) had an average node degree of around 1.

More recently, Matuschek and Gurevych (2013) leveraged monosemous linking (cf. Section 5) in order to create denser semantic graphs for OW and WT. Our approach, however, thanks to the connections obtained through ambiguous words, can provide graphs with significantly higher coverage. As an example, for WT, Matuschek and Gurevych (2013) generated a graph where around 30% of the nodes were in isolation, whereas this number drops to around 5% in our corresponding graph.

These results show that our ontologization approach can be used to obtain dense semantic graph representations of lexical resources, while at the same time preserving a high level of accuracy. Now that all the four resources are transformed into semantic graphs, we move to our alignment experiments.

4.2 Alignment Experiments

4.2.1 Experimental setup

Datasets. As our benchmark we tested on the gold standard datasets used in Matuschek and Gurevych (2013) for three alignment tasks: WordNet-Wikipedia (WN-WP), WordNet-Wiktionary (WN-WT), and WordNet-OmegaWiki (WN-OW). However, the dataset for WN-OW was originally built for the German language and, hence, was missing many English OW concepts that could be considered as candidate target alignments. We therefore fixed the dataset for the English language and reproduced the performance of previous work on the new dataset. The three datasets contained 320, 484, and 315 WN concepts that were manually mapped to their corresponding concepts in WP, WT, and OW, respectively.

Approach	Training type	WN-WP				WN-WT				WN-OW			
		P	R	F1	A	P	R	F1	A	P	R	F1	A
SB	Cross-val.	0.780	0.780	0.780	0.950	0.670	0.650	0.660	0.910	0.749	0.691	0.716	0.886
DWSA	Tuning on subset	0.750	0.670	0.710	0.930	0.680	0.270	0.390	0.890	0.651	0.372	0.473	0.830
SB+DWSA	Cross-val. + tuning	0.750	0.870	0.810	0.950	0.680	0.710	0.690	0.920	0.794	0.688	0.735	0.898
SemAlign	Unsupervised	0.709	0.929	0.805	0.943	0.642	**0.799**	0.712	0.923	0.664	**0.761**	0.709	0.872
	Tuning on subset	**0.877**	0.792	0.833	0.960	0.672	**0.799**	**0.730**	0.930	0.750	0.717	0.733	0.893
	Cross-val.	0.852	0.835	**0.840**	**0.965**	0.680	0.769	0.722	**0.931**	0.778	0.725	**0.749**	**0.900**
	Tuning on WN-WP	-	-	-	-	**0.754**	0.627	0.684	**0.931**	**0.825**	0.584	0.684	0.889
	Tuning on WN-WT	0.738	**0.934**	0.824	0.950	-	-	-	-	0.805	0.677	0.736	**0.900**
	Tuning on WN-OW	0.744	0.925	0.824	0.950	0.684	0.766	0.723	0.930	-	-	-	-

Table 3: The performance of different systems on the task of aligning WordNet to Wikipedia (WN-WP), Wiktionary (WN-WT), and OmegaWiki (WN-OW) in terms of Precision (P), Recall (R), F1, and Accuracy (A). We present results for different configurations of our system (SemAlign), together with the state of the art in definition similarity-based alignment approaches (SB) and the best configuration of the state-of-the-art graph-based system, Dijkstra-WSA (Matuschek and Gurevych, 2013, DWSA).

Configurations. Recall from Section 2 that our resource alignment technique has two parameters: the similarity threshold θ and the combination parameter β, both defined in [0, 1]. We performed experiments with three different configurations:

- *Unsupervised*, where the two parameters are set to their middle values (i.e., 0.5), hence, no tuning is performed for either of the parameters. In this case, both the definitional and structural similarity scores are treated as equally important and two concepts are aligned if their overall similarity exceeds the middle point of the similarity scale.
- *Tuning*, where we follow Matuschek and Gurevych (2013) and tune the parameters on a subset of the dataset comprising 100 items.
- *Cross-validation*, where a 5-fold cross validation is carried out to find the optimal values for the parameters, a technique used in most of the recent alignment methods (Niemann and Gurevych, 2011; Meyer and Gurevych, 2012a; Matuschek and Gurevych, 2013).

4.2.2 Results

We show in Table 3 the alignment performance of different systems on the task of aligning WN-WP, WN-WT, and WN-OW in terms of Precision (P), Recall (R), F1, and Accuracy. The SB system corresponds to the state-of-the-art definition similarity approaches for WN-WP (Niemann and Gurevych, 2011), WN-WT (Meyer and Gurevych, 2011), and WN-OW (Gurevych et al., 2012). DWSA stands for Dijkstra-WSA, the state-of-the-art graph-based alignment approach of Matuschek and Gurevych (2013). The authors also provided results for SB+Dijkstra-WSA, a hybrid system where DWSA was tuned for high precision and, in the case when no alignment target could be found, the algorithm fell back on SB judgments. We also show the results for this system as SB+DWSA in the table.

For our approach (SemAlign) we show the results of six different runs each corresponding to a different setting. The first three (middle part of the table) correspond to the results obtained with the three configurations of SemAlign: unsupervised, with tuning on subset, and cross-validation (see Section 4.2.1). In addition to these, we performed experiments where the two parameters of SemAlign were tuned on pair-independent training data, i.e., a training dataset for a pair of resources different from the one being aligned. For this setting, we used the whole dataset of the corresponding resource pair to tune the two parameters of our system. We show the results for this setting in the bottom part of the table (last three lines).

The main feature worth remarking upon is the consistency in the results across different resource pairs: the unsupervised system gains the best recall among the three configurations (with the improvement over SB+DWSA being always statistically significant[4]) whereas tuning, both on a subset or through cross-validation, consistently leads to the best performance in terms of F1 and accuracy (with the latter being statistically significant with respect to SB+DWSA on WN-WP and WN-WT).

Moreover, the unsupervised system proves to be very robust inasmuch as it provides competitive results on all the three datasets, while it surpasses the performance of SB+DWSA on WN-WT. This

[4] All significance tests are done using z-test at $p < 0.05$.

Approach	**WN-WP**				**WN-WT**				**WN-OW**			
	P	**R**	**F1**	**A**	**P**	**R**	**F1**	**A**	**P**	**R**	**F1**	**A**
Dijkstra-WSA	0.750	0.670	0.710	0.930	**0.680**	0.270	0.390	0.890	0.651	0.372	0.473	0.830
SemAlign$_{str}$	**0.877**	**0.788**	**0.830**	**0.959**	0.604	**0.643**	**0.623**	**0.907**	**0.654**	**0.602**	**0.627**	**0.853**

Table 4: Performance of SemAlign when using only the structural similarity component (SemAlign$_{str}$) compared to the state-of-the-art graph-based alignment approach, Dijkstra-WSA (Matuschek and Gurevych, 2013) for our three resource pairs: WordNet to Wikipedia (WN-WP), Wiktionary (WN-WT), and OmegaWiki (WN-OW).

is particularly interesting as the latter system involves tuning of several parameters, whereas SemAlign, in its unsupervised configuration, does not need any training data nor does it involve any tuning. In addition, as can be seen in the table, SemAlign benefits from pair-independent training data in most cases across the three resource pairs with performance surpassing that of SB+DWSA, a system which is dependent on pair-specific training data. The consistency in the performance of SemAlign in its different configurations and across different resource pairs indicates its robustness and shows that our system can be utilized effectively for aligning any pair of lexical resources, irrespective of their structure or availability of training data.

The system performance is generally higher on the alignment task for WP compared to WT and OW. We attribute this difference to the dictionary nature of the latter two, where sense distinctions are more fine-grained, as opposed to the relatively concrete concepts in the WP encyclopedia.

4.3 Similarity Measure Analysis

We explained in Section 2.1 that our concept similarity measure consists of two components: the definitional and the structural similarities. Measuring the similarity of two concepts in terms of their definitions has been investigated in previous work (Niemann and Gurevych, 2011; Henrich et al., 2012). The structural similarity component of our approach, however, is novel, but at the same time one of the very few measures which enables the computation of the similarity of concepts across two resources directly and independently of the similarity of their definitions. A comparable approach is the Dijkstra-WSA proposed by Matuschek and Gurevych (2013) which, as also mentioned earlier in the Introduction, first connects the two resources' graphs by leveraging monosemous linking and then aligns two concepts across the two graphs on the basis of their shortest distance. To gain more insight into the effectiveness of our structural similarity measure in comparison to the Dijkstra-WSA method, we carried out an experiment where our alignment system used only the structural similarity component, a variant of our system we refer to as SemAlign$_{str}$. Both systems (i.e., SemAlign$_{str}$ and Dijkstra-WSA) were tuned on 100-item subsets of the corresponding datasets.

We show in Table 4 the performance of the two systems on our three datasets. As can be seen in the table, SemAlign$_{str}$ consistently improves over Dijkstra-WSA according to recall, F1 and accuracy with all the differences in recall and accuracy being statistically significant ($p < 0.05$). The improvement is especially noticeable for pairs involving either WT or OW where, thanks to the relatively denser semantic graphs obtained by means of our ontologization technique, the gap in F1 is about 0.23 (WN-WT) and 0.15 (WN-OW).

In addition, as we mentioned earlier, for WN-WP we used the same graph as that of Dijkstra-WSA, since both WN and WP provide a full-fledged semantic network and thus neither needed to be ontologized. Therefore, the considerable performance improvement over Dijkstra-WSA on this resource pair shows the effectiveness of our novel concept similarity measure independently of the underlying semantic network.

5 Related Work

Resource ontologization. Having lexical resources represented as semantic networks is highly beneficial. A good example is WordNet, which has been exploited as a semantic network in dozens of NLP tasks (Fellbaum, 1998). A recent prominent case is Wikipedia (Medelyan et al., 2009; Hovy et al., 2013) which, thanks to its inter-article hyperlink structure, provides a rich backbone for structuring additional information (Auer et al., 2007; Suchanek et al., 2008; Moro and Navigli, 2013; Flati et al., 2014). However, there are many large-scale resources, such as Wiktionary for instance, which by their very nature are not in the form of a graph. This is usually the case with machine-readable dictionaries, where structuring the resource involves the arduous task of connecting lexicographic senses by means of semantic relations. Surprisingly, despite their vast potential, little research has been conducted on the automatic ontologization of collaboratively-constructed dictionaries like Wiktionary and OmegaWiki. Meyer and Gurevych (2012a) and Matuschek and Gurevych (2013) provided approaches for building graph representations of Wiktionary and OmegaWiki. The resulting graphs, however, were either sparse or had a considerable portion of the nodes left in isolation. Our approach, in contrast, aims at transforming a lexical resource into a full-fledged semantic network, hence providing a denser graph with most of its nodes connected.

Resource alignment. Aligning lexical resources has been a very active field of research in the last decade. One of the main objectives in this area has been to enrich existing ontologies by means of complementary information from other resources. As a matter of fact, most efforts have been concentrated on aligning the *de facto* community standard sense inventory, i.e. WordNet, to other resources. These include: the Roget's thesaurus and Longman Dictionary of Contemporary English (Kwong, 1998), FrameNet (Laparra and Rigau, 2009), VerbNet (Shi and Mihalcea, 2005) or domain-specific terminologies such as the Unified Medical Language System (Burgun and Bodenreider, 2001). More recently, the growth of collaboratively-constructed resources has seen the development of alignment approaches with Wikipedia (Ruiz-Casado et al., 2005; Auer et al., 2007; Suchanek et al., 2008; Reiter et al., 2008; Navigli and Ponzetto, 2012), Wiktionary (Meyer and Gurevych, 2011) and OmegaWiki (Gurevych et al., 2012). Last year Matuschek and Gurevych (2013) proposed Dijkstra-WSA, a graph-based approach relying on shortest paths between two concepts when the two corresponding resources graphs were combined by leveraging monosemous linking. Their method when backed off with other definition similarity based approaches (Niemann and Gurevych, 2011; Meyer and Gurevych, 2011), achieved state-of-the-art results on the mapping of WordNet to different collaboratively-constructed resources. This approach, however, in addition to setting the threshold for the definition similarity component by means of cross validation, also required other parameters to be tuned, such as the allowed path length (λ) and the maximum number of edges in a graph. The optimal value for the λ parameter varied from one resource pair to another, and even for a specific resource pair it had to be tuned for each configuration. This made the approach dependent on the training data for the specific pair of resources that were to be aligned. Instead of measuring the similarity of two concepts on the basis of their distance in the combined graph, our approach models each concept through a rich vectorial representation we refer to as semantic signature and compares the two concepts in terms of the similarity of their semantic signatures. This rich representation leads to our approach having a good degree of robustness such that it can achieve competitive results even in the absence of training data. This enables our system to be applied effectively for aligning new pairs of resources for which no training data is available, with state-of-the-art performance.

6 Conclusions

This paper presents a unified approach for aligning lexical resources. Our method leverages a novel similarity measure which enables a direct structural comparison of concepts across different lexical resources. Thanks to an effective ontologization method, our alignment approach can be applied to any pair of lexical resources independently of whether they provide a full-fledged network structure. We demonstrate that our approach achieves state-of-the-art performance on aligning WordNet to three collaboratively-constructed resources with different characteristics, i.e., Wikipedia, Wiktionary, and OmegaWiki. We also show that our approach is robust across its different configurations, even when the training data is absent, enabling it to be used effectively for aligning new pairs of lexical resources for which no resource-specific training data is available. In future work, we plan to extend our concept similarity measure across different natural languages. We release all our data at `http://lcl.uniroma1.it/semalign`.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Starting Grant MultiJEDI No. 259234.

We would like to thank Michael Matuschek for providing us with Wikipedia graphs and alignment datasets.

References

Eneko Agirre and Aitor Soroa. 2009. Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 33–41, Athens, Greece.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ive. 2007. DBpedia: A nucleus for a web of open data. In *Proceedings of 6th International Semantic Web Conference joint with 2nd Asian Semantic Web Conference (ISWC+ASWC 2007)*, pages 722–735, Busan, Korea.

Sergey Brin and Michael Page. 1998. Anatomy of a large-scale hypertextual Web search engine. In *Proceedings of the 7^{th} Conference on World Wide Web*, pages 107–117, Brisbane, Australia.

Anita Burgun and Olivier Bodenreider. 2001. Comparing terms, concepts and semantic classes in WordNet and the Unified Medical Language System. In *Proceedings of NAACL Workshop, WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 77–82, Pittsburgh, USA.

Gerard de Melo and Gerhard Weikum. 2010. Providing multilingual, multimodal answers to lexical database queries. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 348–355, Valletta, Malta.

Stefano Faralli and Roberto Navigli. 2012. A New Minimally-supervised Framework for Domain Word Sense Disambiguation. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1411–1422, Jeju, Korea.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Database*. MIT Press, Cambridge, MA.

Tiziano Flati, Daniele Vannella, Tommaso Pasini, and Roberto Navigli. 2014. Two is bigger (and better) than one: the Wikipedia Bitaxonomy Project. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, Baltimore, Maryland.

Iryna Gurevych, Judith Eckle-Kohler, Silvana Hartmann, Michael Matuschek, Christian M. Meyer, and Christian Wirth. 2012. UBY - a large-scale unified lexical-semantic resource based on LMF. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 580–590, Avignon, France.

Taher H. Haveliwala. 2002. Topic-sensitive PageRank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526, Hawaii, USA.

Verena Henrich, Erhard Hinrichs, and Tatiana Vodolazova. 2011. Semi-automatic extension of GermaNet with sense definitions from Wiktionary. In *Proceedings of 5th Language & Technology Conference (LTC 2011)*, pages 126–130, Pozna, Poland.

Verena Henrich, Erhard W. Hinrichs, and Klaus Suttner. 2012. Automatically linking GermaNet to Wikipedia for harvesting corpus examples for GermaNet senses. *In Journal for Language Technology and Computational Linguistics (JLCL)*, 27(1):1–19.

Eduard H. Hovy, Roberto Navigli, and Simone Paolo Ponzetto. 2013. Collaboratively built semi-structured content and Artificial Intelligence: The story so far. *Artificial Intelligence*, 194:2–27.

Thad Hughes and Daniel Ramage. 2007. Lexical semantic relatedness with random graph walks. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing*, pages 581–589, Prague, Czech Republic.

Oi Yee Kwong. 1998. Aligning WordNet with additional lexical resources. In *COLING-ACL98 Workshop on Usage of WordNet in Natural Language Processing Systems*, pages 73–79, Montreal, Canada.

Egoitzand Laparra and German Rigau. 2009. Integrating WordNet and FrameNet using a knowledge-based Word Sense Disambiguation algorithm. In *Proceedings of Recent Advances in Natural Language Processing (RANLP09)*, pages 1–6, Borovets, Bulgaria.

Michael Matuschek and Iryna Gurevych. 2013. Dijkstra-WSA: A graph-based approach to word sense alignment. *Transactions of the Association for Computational Linguistics (TACL)*, 1:151–164.

Olena Medelyan, David Milne, Catherine Legg, and Ian H. Witten. 2009. Mining meaning from Wikipedia. *International Journal of Human-Computer Studies*, 67(9):716–754.

Christian M. Meyer and Iryna Gurevych. 2010. "worth its weight in gold or yet another resource"; a comparative study of Wiktionary, OpenThesaurus and GermaNet. In *Proceedings of the 11th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing'10, pages 38–49, Iasi, Romania.

Christian M. Meyer and Iryna Gurevych. 2011. What psycholinguists know about Chemistry: Aligning Wiktionary and WordNet for increased domain coverage. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 883–892, Chiang Mai, Thailand.

Christian M. Meyer and Iryna Gurevych. 2012a. OntoWiktionary: Constructing an ontology from the collaborative online dictionary Wiktionary. In *Semi-Automatic Ontology Development: Processes and Resources*, pages 131–161. IGI Global.

Christian M. Meyer and Iryna Gurevych. 2012b. To exhibit is not to loiter: A multilingual, sense-disambiguated Wiktionary for measuring verb similarity. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 1763–1780, Mumbai, India.

Andrea Moro and Roberto Navigli. 2013. Integrating syntactic and semantic analysis into the Open Information Extraction paradigm. In *Proceedings of the 23^{rd} International Joint Conference on Artificial Intelligence (IJCAI 2013)*, pages 2148–2154, Beijing, China.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Roberto Navigli, Stefano Faralli, Aitor Soroa, Oier de Lacalle, and Eneko Agirre. 2011. Two birds with one stone: Learning semantic models for text categorization and Word Sense Disambiguation. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management (CIKM)*, pages 2317–2320, Glasgow, UK.

Roberto Navigli. 2006. Meaningful clustering of senses helps boost word sense disambiguation performance. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics joint with the 21st International Conference on Computational Linguistics (COLING-ACL 2006)*, pages 105–112, Sydney, Australia.

Elisabeth Niemann and Iryna Gurevych. 2011. The people's web meets linguistic knowledge: Automatic sense alignment of Wikipedia and WordNet. In *Proceedings of the Ninth International Conference on Computational Semantics*, pages 205–214, Oxford, United Kingdom.

Martha Palmer, Daniel Gildea, and Nianwen Xue. 2010. *Semantic Role Labeling*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Patrick Pantel and Marco Pennacchiotti. 2008. Automatically harvesting and ontologizing semantic relations. In *Proceedings of the 2008 Conference on Ontology Learning and Population: Bridging the Gap Between Text and Knowledge*, pages 171–195, Amsterdam, The Netherlands.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, Disambiguate and Walk: a Unified Approach for Measuring Semantic Similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1341–1351, Sofia, Bulgaria.

Nils Reiter, Matthias Hartung, and Anette Frank. 2008. A resource-poor approach for linking ontology classes to Wikipedia articles. In Johan Bos and Rodolfo Delmonte, editors, *Semantics in Text Processing*, volume 1 of *Research in Computational Semantics*, pages 381–387. College Publications, London, England.

Maria Ruiz-Casado, Enrique Alfonseca, and Pablo Castells. 2005. Automatic assignment of Wikipedia encyclopedic entries to WordNet synsets. In *Proceedings of the Third International Conference on Advances in Web Intelligence*, pages 380–386, Lodz, Poland.

Lei Shi and Rada Mihalcea. 2005. Putting pieces together: Combining FrameNet, VerbNet and WordNet for robust semantic parsing. In *Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 100–111, Mexico City, Mexico.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. Yago: A large ontology from Wikipedia and WordNet. *Journal of Web Semantics*, 6(3):203–217.

Torsten Zesch, Christof Müller, and Iryna Gurevych. 2008. Using Wiktionary for computing semantic relatedness. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*, pages 861–866, Chicago, Illinois.