Latent semantic network induction in the context of linked example senses

Hunter Scott Heidenreich
Department of Computer Science
College of Computing and Informatics
hsh28@drexel.edu

Jake Ryland Williams
Department of Information Science
College of Computing and Informatics
jw3477@drexel.edu

Abstract

The Princeton WordNet is a powerful tool for studying language and developing natural language processing algorithms. With significant work developing it further, one line considers its extension through aligning its expert-annotated structure with other lexical resources. In contrast, this work explores a completely data-driven approach to network construction, forming a wordnet using the entirety of the open-source, noisy, user-annotated dictionary, Wiktionary. Comparing baselines to WordNet, we find compelling evidence that our network induction process constructs a network with useful semantic structure. With thousands of semantically-linked examples that demonstrate sense usage from basic lemmas to multiword expressions (MWEs), we believe this work motivates future research.

1 Introduction

Wiktionary is a free and open-source collaborative dictionary[1] (Wikimedia). With the ability for anyone to add or edit lemmas, definitions, relations, and examples, Wiktionary has the potential to be larger and more diverse than any printable dictionary. Wiktionary features a rich set of examples of sense usage for many of its lemmas which, when converted to a usable format, supports language processing tasks such as sense disambiguation (Meyer and Gurevych, 2010a; Matuschek and Gurevych, 2013; Miller and Gurevych, 2014) and MWE identification (Muzny and Zettlemoyer, 2013; Salehi et al., 2014; Hosseini et al., 2016). With natural alignment to other languages, Wiktionary can likewise be used as a resource for machine translation tasks (Matuschek et al., 2013; Borin et al., 2014; Göhring, 2014). With these uses in mind, this work introduces the creation of a network—much like the Princeton WordNet (Miller, 1995; Fellbaum, 1998)—that is constructed solely from the semi-structured data of Wiktionary. This relies on the noisy annotations of the editors of Wiktionary to naturally induce a network over the entirety of the English portion of Wiktionary. In doing so, the development of this work produces:

- an induced network over Wiktionary, enriched with semantically linked examples, forming a directed acyclic graph (DAG);
- an exploration of the task of relationship disambiguation as a means to induce network construction; and
- an outline for directions of expansion, including increasing precision in disambiguation, cross-linking example usages, and aligning English Wiktionary with other languages.

We make our code freely available[2], which includes code to download data, to disambiguate relationships between lemmas, to construct networks from disambiguation output, and to interact with networks produced through this work.

2 Related work

2.1 WordNet

The Princeton WordNet, or WordNet as it's more commonly referred to, is a lexical database originally created for the English language (Miller, 1995; Fellbaum, 1998). It consists of expert-annotated data, and has been more or less continually updated since its creation (Harabagiu et al., 1999; Miller and Hristea, 2006). WordNet is built up of *synsets*, collections of lexical items that all

[1] https://www.wiktionary.org/

[2] Code will be available at https://github.com/hunter-heidenreich/lsni-paper

Proceedings of the 2019 EMNLP Workshop W-NUT: The 5th Workshop on Noisy User-generated Text, pages 170–180
Hong Kong, Nov 4, 2019.

have the same meaning. For each synset, a definition is provided, and for *some* synsets, usage examples are also presented. If extracted and attributed properly, the example usages present on Wiktionary could critically enhance WordNet by filling gaps. While significant other work has been done in utilizing Wiktionary to enhance WordNet for purposes like this (discussed in the next sections), this work takes a novel step by constructing a wordnet through entirely computational means, i.e. under the framing of a machine learning task based on Wiktionary's data.

2.2 Wiktionary

Wiktionary is an open-source, Wiki-based, open content dictionary organized by the WikiMedia Foundation (Wikimedia). It has a large and active volunteer editorial community, and from its noisy, crowd-sourced nature, includes *many* MWEs, colloquial terms, and their example usages, which could ultimately fill difficult-to-resolve gaps left in other linguistic resources, such as WordNet.

Thus, Wiktionary has a significant history of exploration for the enhancement of WordNet, including efforts that extend WordNet for better domain coverage of word senses (Meyer and Gurevych, 2011; Gurevych et al., 2012; Miller and Gurevych, 2014), automatically derive new lemmas (Jurgens and Pilehvar, 2015; Rusert and Pedersen, 2016), and develop the creation of multilingual wordnets (de Melo and Weikum, 2009; Gurevych et al., 2012; Bond and Foster, 2013). While these works constitute important steps in the usage of *extracted* Wiktionary contents for the development of WordNet, none before this effort has attempted to utilize the *entirety* of Wiktionary alone for the construction of such a network.

Most similarly, Wiktionary has been used in a sense-disambiguated fashion (Meyer and Gurevych, 2012b) and to construct an ontology (Meyer and Gurevych, 2012a). Our work does not create an ontology, but instead attempts to create a semantic wordnet. In this context, our work can be viewed as building on notions of sense-disambiguating Wiktionary to construct a WordNet-like resource.

2.3 Relation Disambiguation

The task of taking definitions, a semantic relationship, and sub-selecting the definitions that belong to that relationship is one of critical importance to our work. Sometimes called sense linking or relationship anchoring, this task has been previously explored in the creation of machine-readable dictionaries (Krovetz, 1992), ontology learning (Pantel and Pennacchiotti, 2006, 2008), and German Wikitionary (Meyer and Gurevych, 2010b).

As mentioned above, Meyer and Gurevych explore relationship disambiguation in the context of Wiktionary, motivating a sense-disambiguated Wiktionary as a powerful resource (Meyer and Gurevych, 2012a,b). This task is frequently viewed as a binary classification: Given two linked lemmas, do these pairs of definitions belong to the relationship? While easier to model, this framing can suffer from a combinatorial explosion as all pairs of definitions must be compared. This work attempts to model the task differently, disambiguating all definitions in the context of a relationship and its lemmas.

3 Model

3.1 Framework

This work starts by identifying a set of lemmas, W, and a set of senses, S. It then proceeds, assuming that S forms the vertex set of a Directed Acyclic Graph (DAG) with edge set E, organizing S by refinement of specificity. That is, if senses $s, t \in S$ have a link $(t, s) \in E$—*to* s—then s is one degree of refinement more specific than t.

Next, we suppose a lemma $u \in W$ has relation $\sim$ (e.g., synonymy) indicated to another lemma $v \in W$. Assuming $\sim$ is recorded from u to v (e.g., from u's page), we call u the source and v the sink. Working along these lines, the model then assumes a given indicated relation $\sim$ is qualified by a sense s; this semantic equivalence is denoted $u \overset{s}{\sim} v$.

Like others (Landauer and Dumais, 1997; Blei et al., 2003; Bengio et al., 2003), this work assumes senses exist in a latent semantic space. Processing a dictionary, one can empirically discover relationships like $u \overset{s}{\sim} v$ and $v \overset{t}{\sim} w$. But for a larger network structure one must know if $s = t$—that is, do s and t refer to the same relationship—and often neither s nor t are known, explicitly. Hence, this work sets up approximations of s and t for comparison. Given a lemma, $u \in W$, suppose a set of definitions, D_u, exists and form the basis for disambiguation of a lemma's senses. We then assume that for any $d \in D_u$ there exists one or more senses, $s \in S$, such that $d \implies s$, that is, the definition d conveys the sense s.

Having assumed a DAG structure for S, this work denotes specificity of sense by using the formalism of a partial order, $\preceq$, which, for senses $s, t \in S$ having $s \preceq t$, indicates that the sense s is comparable to t and more specific. Note that—as with any partial order—senses can be, and are often non-comparable.

Intuitively, a given definition d might convey multiple senses $d \implies s, t$ of differing specificities, $s \preceq t$. So for a given definition d, the model's goal is to find the sense t that is least specific in being conveyed. Satisfying this goal implies resolving the sense identification function, $f : D \to S$, for which any lemma $u \in W$ and definition $d \in D_u$ with $d \implies s \in S$, it is assured that $s \preceq f(d)$. Since no direct knowledge of any $s \in S$ is assumed known for any annotated relationship between lemmas, systems must approximate senses according to the available resources, e.g., definitions or example usages.

3.2 Task development

On Wiktionary, every lemma has its own page. Each page is commonly broken down into sections such as languages, etymologies, and parts-of-speech (POS). Under each POS, a lemma features a set of definitions that can be automatically extracted. An example of the word *induce* on English Wiktionary can be seen in Figure 1.

A significant benefit of using Wiktionary as a resource to build a wordnet lies in the wealth of examples it offers. Examples come in two flavors: basic usage and usage from reference material. Currently, each example is linked to its origination definition and lemma, however, in future works, these examples could be segmented and sense disambiguated, offering new network links and densely connected example usages.

For each lemma, Wiktionary may offer relationship annotations between lemmas. These relationships span many categories including acronyms, alternative forms, anagrams, antonyms, compounds, conjugations, derived terms, descendants, holonyms, hypernyms, hyponyms, meronyms, related terms, and synonyms. For this work's purposes, only antonyms and synonyms are considered, exploiting their more typical structure on Wiktionary and clear theoretical basis in semantic equivalence to induce a network. Exploring more of these relationships is of interest in future work.

Additionally, a minority of annotations present 'gloss' labels, which indicate the definitions that apply to relationships. So from the data there is some knowledge of exact matching, but due to their limited, noisy, and crowd-sourced nature, the labelings may not cover all definitions that belong.

We assume annotations exhibit relationships between lemmas. Finding one: $u \overset{s}{\sim} v$, if u is the source, we assume there exists some definition $d \in D_u$ that implies the appropriate sense: $d \implies s$. This good practice assumption models editor behavior as a response to exposure to a particular definition on the source page. Provided this, an editor won't necessarily annotate the relationship on the sink page—even if the sink page has a definition that implies the sense s. Thus, our task doesn't *require* identification of a definition on the sink's page. More precisely, no $d \in D_v$ might exist that implies s ($d \implies s$) for an annotated relationship, $u \overset{s}{\sim} v$.

Altogether, for an annotated relationship the task aims to identify the sense-conveying subset:

$$D_{u \overset{s}{\sim} v} = \{d \in D_u \cup D_v \mid d \implies s\}$$

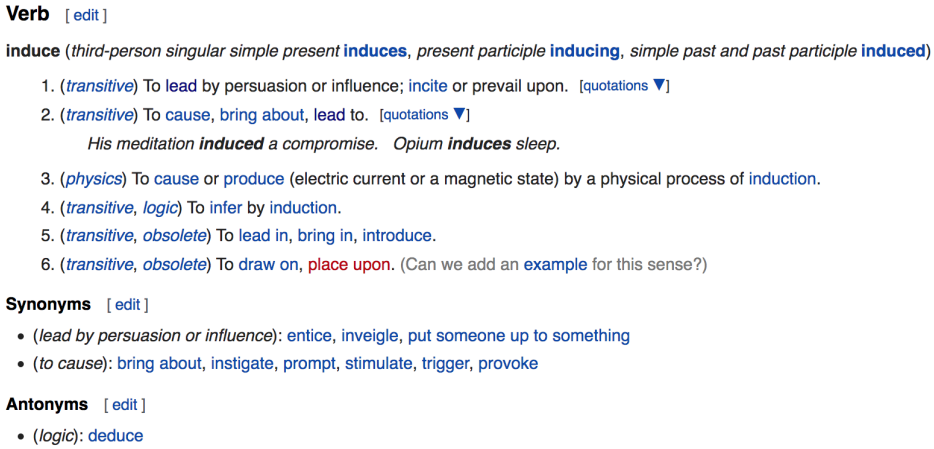
for which at least one definition must be drawn from D_u. Note that the model *does not* assume that arbitrary $d, \tilde{d} \in D_{u \overset{s}{\sim} v}$ map through the sense identification function to the same most general sense. Presently, these details are resolved by a separate algorithm (developed below), leaving direct modeling of the sense identification function to future work.[3]

3.3 Semantic hierarchy induction

This section outlines preliminary work inferring a semantic hierarchy from pairwise relationships. If A is the set of relationships, a model's output, C, will be a collection of sense-conveying subsets, $D_{u \overset{s}{\sim} v}$, in one-to-one correspondence: $A \leftrightarrow C$. So, for all $\mathbb{D} \in \mathcal{P}(C)$, one has a covering of (some) senses by pairwise relationships, $D_{u \overset{s}{\sim} v} \in \mathbb{D}$.

Under our assumptions, any collection of sense conveying subsets $\mathbb{D} \in \mathcal{P}(C)$ with non-empty intersection restricts to a set of definitions that must convey at least one common sense, s'. Notably, s' must be at least as general as any qualifying a particular annotated relationship, i.e., $s \preceq s'$ for any s (implicitly) defining any $D_{u \overset{s}{\sim} v} \in \mathbb{D}$.

[3] A major challenge to this approach is the increased complexity required for the development of evaluation data.



Verb [edit]

induce (*third-person singular simple present* **induces**, *present participle* **inducing**, *simple past and past participle* **induced**)

1. (*transitive*) To lead by persuasion or influence; incite or prevail upon. [quotations ▼]
2. (*transitive*) To cause, bring about, lead to. [quotations ▼]
   *His meditation **induced** a compromise. Opium **induces** sleep.*
3. (*physics*) To cause or produce (electric current or a magnetic state) by a physical process of induction.
4. (*transitive*, *logic*) To infer by induction.
5. (*transitive*, *obsolete*) To lead in, bring in, introduce.
6. (*transitive*, *obsolete*) To draw on, place upon. (Can we add an example for this sense?)

Synonyms [edit]

- (*lead by persuasion or influence*): entice, inveigle, put someone up to something
- (*to cause*): bring about, instigate, prompt, stimulate, trigger, provoke

Antonyms [edit]

- (*logic*): deduce

Figure 1: The Verb section of the induce page on English Wiktionary. Definitions are enumerated, with example usages as sub-elements or drop-down quotations. Relationships for this page are well annotated, with gloss labels to indicate the definition that prompted annotation.

So this work induces the sense-identification function, f, through pre-images: for $\mathbb{D} \in \mathcal{P}(C)$, an implicit sense, s, is assumed such that that $f^{-1}(s) \subseteq \bigcap_{\mathbb{D}} D_{u \overset{s}{\sim} v}$. Now, if a covering $\mathbb{D}' \supset \mathbb{D}$ exists with non-empty intersection, then its (smaller) intersection comprises definitions that convey a sense, s' which is more-general than s. So to precisely resolve f through pre-images the model must 'hole punch' the more-general definitions, constructing the hierarchy by allocating the more general definitions in the intersection of $\mathbb{D}'$ to the more general senses:

$$f^{-1}(t) = \left(\bigcap_{\mathbb{D}} D_{u \overset{s}{\sim} v} \right) \setminus \left(\bigcap_{\mathbb{D}' \supset \mathbb{D}} \bigcap_{\mathbb{D}'} D_{u' \overset{s'}{\sim} v'} \right).$$

This allocates each definition to exactly one implicit sense approximation, t, which is the most general sense indicated by the definition. Additionally, all senses then fall under a DAG hierarchy (excepting the singletons, addressed below) as set inclusion, $\mathbb{D}' \supset \mathbb{D}$ defines a partial order. This deterministic algorithm for hierarchy induction is presented in Algorithm 1.

Considering the output of a model, C, if d is not covered by C the model assumes a singleton sense. These include definitions not selected during relationship disambiguation as well as the definitions of lemmas that feature no relationship annotations. Singletons are then placed in the DAG at the lowest level, disconnected from all other senses. Figure 2 visually represents this full semantic hierarchy.

Algorithm 1 Construction of semantic hierarchy through pairwise collection.

Require: C: Collection of $D_{u \overset{s}{\sim} v}$
  $levels \leftarrow List()$
  $prev \leftarrow C$
  while $prev \neq \emptyset$ **do**
    $next \leftarrow List()$
    $defs \leftarrow \emptyset$
    for $p, p' \in prev$ **do**
      if $p \neq p'$ and $p \cap p' \neq \emptyset$ **then**
        $Append(next, p \cap p')$
        $Union(defs, p \cap p')$
      end if
    end for
    $filtered \leftarrow List()$
    for $p \in prev$ **do**
      $Append(filtered, p \setminus defs)$
    end for
    $Append(levels, filtered)$
    $prev \leftarrow next$
  end while
  return $levels$

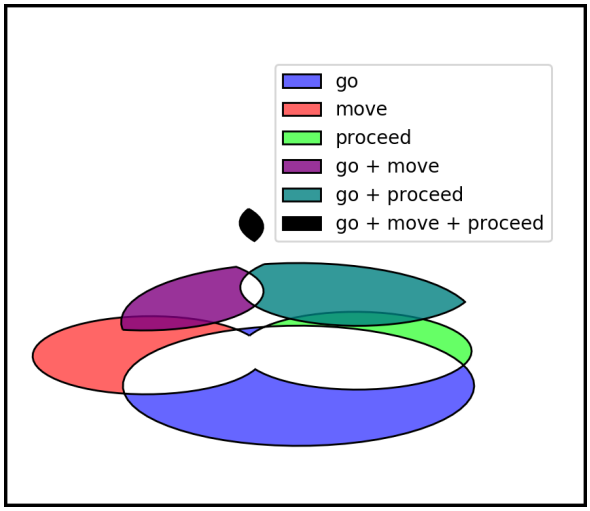


Figure 2: A visualization of 3 lemmas intersecting to create a semantic hierarchy.

4 Evaluation

4.1 Characteristics of Wiktionary data

Data was downloaded from Wiktionary on 1/23/19 using the Wikimedia Rest API[4]. To evaluate performance, a 'gold' dataset was created to compare modeling strategies. In totality, 298,377 synonym and 44,758 antonym links were generated from Wiktionary. 'Gold' links were randomly sampled, selecting 400 synonym and 100 antonym links. For each link, source and sink lemmas were considered independently. Definitions were included if they could plausibly refer to the other lemma. This process is supported by the available examples, testing if one lemma can replace the other lemma in the example usages. This dataset was constructed in contrast to other Wiktionary relationship disambiguation tasks due to the modeling differences and desire for more synonym- and antonym-specific evaluations (Meyer and Gurevych, 2012a,b).

4.2 Evaluation strategy

This work's evaluation considers precision, recall, and variants of the F_β score (biasing averages of precision and recall). As there is selection on both source and sink sides, we consider several averaging schemes. For a final evaluation, each sample is averaged at the side-level and averaged across all relationships. Macro-averages compute an unweighted average, while micro-averages weight performance based on the number of definitions involved in the selection process. Intuitively, micro metrics weight based on size, while macro metrics ignore size (treating all potential links and sides as equal).

4.3 Setting up baselines

For baselines, we present two types of models, which we refer to as *return all* and *vector similarity*. The return all baseline model assumes that for a given relationship link, all definitions belong. This is not intended as a model that could produce a useful network as many definitions and lemmas would be linked that clearly do not belong together. This achieves maximum recall at the expense of precision, demonstrating a base level of precision that must be exceeded.

The vector similarity baseline model takes advantage of semantic vector representations for computing similarity (Bengio et al., 2003; Mikolov et al., 2013; Pennington et al., 2014; Joulin et al., 2017). It computes the similarity between lemmas and definitions, utilizing thresholds that flag to either retain similarities above (max), below (min), or with magnitude above the threshold (abs).

Wiktionary features many MWEs and uncommon lemmas requiring use of a vectorization strategy that allows for handling of lemmas not observed in the representation's training. Thus, FastText was selected for its ability to represent out-of-vocabulary lemmas through its bag-of-character n-gram modeling (Bojanowski et al., 2017). To compute similarity between lemmas and definitions, this model aggregates word vectors of the individual tokens present in a definition. Following other work (Lilleberg et al., 2015; Wu et al., 2018), TF-IDF weighted averages of word vectors were utilized in a very simple averaging scheme.

Initial results indicated that a simple cosine similarity with a linear kernel performed marginally above the return all baseline[5]. Thus, kernel tricks (Cristianini and Shawe-Taylor, 2000) were explored (to positive effect). The Gaussian kernel is often recommended as a good initial kernel to try as a baseline (Schölkopf et al., 1995; Joachims,

[4] https://en.wikipedia.org/api/rest_v1/

[5] This is interesting to note, since previous work has found that word embeddings like GloVe and word2vec contain a surprising amount of word frequency effects that pollute simple cosine similarity (Schnabel et al., 2015). This may explain why vanilla cosine similarity performed poorly with FastText vectors here and provides more evidence against using it as the default similarity measure.

1998). It is formulated using a radial basis function (RBF), only dependent on a measure of distance. The Laplacian kernel is a slight variation of the Gaussian kernel, measuring distance as the L1 distance where the Gaussian measures distance as L2 distance. Both kernels fall in the RBF category with a single regularization parameter, γ, and were used in comparison to cosine similarity.

For these kernels, a grid search over γ was conducted from 10^{-3} to 10^{3} at steps of powers of 10. Similarly, similarity comparison thresholds were considered from -1.0 to 1.0 at steps of 0.05 for all 3 thresholding schemes (min, max, abs).

When selecting a final model, F_1 scores were not considered as recall scores outweighed precision under a simple harmonic mean. This resulted in models with identical performance to the return all model or worse. Instead, models were considered against full-precision and $F_{0.1}$ scores.

4.4 Semantic Structure Correlation

Creating a wordnet solely from Wiktionary's noisy, crowd-sourced data begs the question: Does the generated network structure resemble the structure present in Princeton's WordNet? To get a sense of this, we compare the capacities of each of these resources as a basis for semantic similarity modeling (using Pearson correlation (Pearson, 1895)). This work considers three notions of graph-based semantic similarity that are present in WordNet: path similarity (PS), Leacock Chodorow similarity (LCH) (Leacock and Chodorow, 1998), and Wu Palmer similarity (WP) (Wu and Palmer, 1994).

The point of this experiment is *not* to enforce a notion that this network should mirror the structure of WordNet. Given Wiktionary's size, it likely possesses a great deal of information not represented by WordNet (resolved our other experiment on word similarity, Sec. 5.3). But if there is some association between the semantic representation capacities of these two networks we may possibly draw some insight into a more basic question: "has this model produced *some* relevant semantic structure?"

For this experiment, only nouns and verbs are considered as they are the only POS for which WordNet defines these metrics. Additionally, these metrics are defined at the synset level. There is no direct mapping between synsets in our network and WordNet, therefore, scores are considered at a lemma level. By computing values of all pairs of synsets between lemmas, three values per metric are generated: minimum, maximum, and average. Additionally, only lemmas that differ in minimum and maximum similarity are retained, restricting the experiment to the most polysemous portions of the networks.

5 Results

5.1 Baseline model performance

Table 1 shows baseline model performance on the relationship disambiguation task and highlights model parameters. During evaluation, the Laplacian kernel was found to consistently outperform the Gaussian kernel. For this reason, this work presents the scores from the return all baseline and two variants of the Laplacian kernel model—one optimized for precision and the other for $F_{0.1}$.

Note that in the synonym case, max-threshold selection performed best, while in the antonym case min- and abs-threshold fared better. This aligns well with the notion that while synonyms are semantically similar, antonyms are semantically anti-similar—an interesting consideration for future model development.

Overall, from the scores in Table 1 one can see that the vector similarity models improve over the return all, but that there is much work to be done to further improve precision and recall.

5.2 Comparison against WordNet

WordNet publishes several statistics[6] that one can use for quantitative comparison with the network constructed herein. Reviewing the count statistics shows that Wiktionary is an order of magnitude larger than WordNet and that Wiktionary features 344,789 linked example usages to WordNet's 68,411.

Polysemy. Table 2 report polysemy statistics. Despite the difference in creation processes, the induced networks do not have polysemy averages drastically different from WordNet.

In comparing the three networks induced, there is a common theme of increase in polysemy when shifting from recall to precision. This makes sense due to the fact that the return all model will merge all possible lemmas that overlap in relationship annotations resulting in lower polysemy statistics,

[6] Statistics are taken from WordNet's website for WordNet 3.0, last accessed on 8/11/2019: https://wordnet.princeton.edu/documentation/wnstats7wn

Model	Synonyms					Antonyms				
	Thresh.	Recall		Precision		Thresh.	Recall		Precision	
		Macro	Micro	Macro	Micro		Macro	Micro	Macro	Micro
Ret. All		1.000	1.000	0.602	0.268		1.000	1.000	0.527	0.280
Precision	$max_{0.35}$	0.433	0.258	0.847	0.541	$min_{-0.35}$	0.266	0.196	0.820	0.600
$F_{0.1}$	$max_{0.30}$	0.535	0.404	0.814	0.532	$abs_{0.25}$	0.730	0.763	0.619	0.397

Table 1: Model performance with threshold selection. All $\gamma = 0.1$, except for antonym precision where $\gamma = 100$.

whereas a precision-based model will result in pair-wise clusters that do not overlap as broadly, resulting in more complex hierarchies.

Structural differences. Intentionally, the presented notion of a semantic hierarchy functions similarly to the hypernym connections within WordNet. Moving up the semantic hierarchy produces sense approximations from definitions that are more general, and moving down the hierarchy produces more specific senses. However, in the induced networks, this is a notion applied to every POS—WordNet only produces these connections for nouns and verbs. An example taken from the $F_{0.1}$ network is that of the adjective *good* (referring to Holy) being subsumed by a synset featuring the adjective *proper* (referring to suitable, acceptable, and following the established standards).

5.3 Word Similarity

In previous works, WordNet and Wiktionary have been used to create vector representations of words. A common method for evaluating the quality of word vectors is performance on word similarity tasks. Performance on these tasks is evaluated through Spearman's rank correlation (Spearman, 2010) between cosine similarity of vector representations and human annotations.

Using Explicit Semantic Analysis (ESA), a technique based on concept vectors, our network constructs vectors using a word's tf-idf scores over concepts, as has been done in prior works (Gabrilovich and Markovitch, 2007; Zesch et al., 2008; Meyer and Gurevych, 2012b). We define our concepts as senses of the $F_{0.1}$ network and compute cosine similarity in this representation.

We compare performance against other ESA methods (Zesch et al., 2008; Meyer and Gurevych, 2012b) on common datasets: Rubenstein and Goodenough's 65 noun pairs (1965, RG-65), Miller and Charles's 30 noun pairs (1991, MC-30), Finklestein et. al's 353 word similarity pairs (2002, WS-353, split into Fin-153 and Fin-200 due to different annotators), and Yang and Powers's 130 verb pairs (2006, YP-130). Our results are summarized in Table 3.

We also compare $F_{0.1}$ against latent word vector representations like word2vec's continuous bag-of-words (CBOW) and skip-grams (SG) (Mikolov et al., 2013), GloVe (Pennington et al., 2014), and FastText (Bojanowski et al., 2017). These results are presented in Table 4.

In analyzing these results, the $F_{0.1}$ network performs well. Against other ESA methods, it is highly competitive, achieving the highest performance in two datasets. When strictly comparing performance against ESA with WordNet as the source, it has approximately equal or better performance in all datasets except YP-130. We hypothesize that this is due to a lack of precision in verb disambiguation, reinforced by the low polysemy seen above. Additionally, the work from Zesch et al. (2008) evaluated on subsets of the data in which all three resources had coverage. In their work, YP-130 performance is computed for only 80 of the 130 pairs.

Comparing $F_{0.1}$ to latent word vectors, it has the highest performance on noun datasets and is competitive on WS-353. While not directly comparable, it achieves this through 26 million tokens of structured text in contrast to billions of tokens of unstructured text that train latent vectors.

5.4 Network Correlation Results

Table 5 displays correlation values between graph-based semantic similarity metrics of $F_{0.1}$ and WordNet. Pairs of 1,009 verb and 1,303 noun lemmas were considered. In generating similarities, disconnected lemma pairs were discarded, producing 31,373 verb and 16,530 noun pairs. The table shows that for nouns, the two networks produce similarity values that are weakly to moderately correlated, however, verbs produce values that are, at most, very weakly correlated, if at all.

Due to the fact that $F_{0.1}$ produced better results

POS	With Monosemous Words				Without Monosemous Words			
	WordNet	$F_{0.01}$	Precision	Return All	WordNet	$F_{0.01}$	Precision	Return All
Noun	1.24	1.17	1.18	1.10	2.79	2.94	2.99	2.66
Verb	2.17	1.20	1.22	1.10	3.57	3.18	3.33	2.78
Adjective	1.18	1.18	1.18	1.10	2.71	2.59	2.62	2.33
Adverb	1.25	1.11	1.12	1.08	2.50	2.34	2.36	2.25

Table 2: Average polysemy statistics.

Dataset	RG-65	MC-30	Fin-153	Fin-200	YP-130
$F_{0.1}$	0.831	**0.849**	**0.723**	0.557	0.687
WordNet* (Zesch et al., 2008)	0.82	0.78	0.61	0.56	0.71
Wikipedia* (Zesch et al., 2008)	0.76	0.68	0.70	0.50	0.29
Wiktionary* (Zesch et al., 2008)	**0.84**	0.84	0.70	**0.60**	0.65
Wiktionary (Meyer and Gurevych, 2012b)	-	-	-	-	**0.73**

Table 3: Spearman's rank correlation coefficients on word similarity tasks. Best values are in bold.

Dataset	RG-65	MC-30	WS-353
$F_{0.1}$	**0.831**	**0.849**	0.669
FastText	-	-	0.73
CBOW (6B)	0.682	0.656	0.572
SG (6B)	0.697	0.652	0.628
GloVe (6B)	0.778	0.727	0.658
GloVe (42B)	0.829	0.836	**0.759**
CBOW (100B)	0.754	0.796	0.684

Table 4: Spearman's correlation on word similarity tasks. Best values are in bold. Number of tokens in training data is featured in parentheses, if reported. FastText is reported from (Bojanowski et al., 2017), and all others are from (Pennington et al., 2014).

	Noun	Verb
PS min	0.266	0.132
PS max	0.495	0.189
PS avg	0.448	0.082
LCH min	0.207	0.120
LCH max	0.384	0.056
LCH avg	0.359	-0.013
WP min	0.116	0.090
WP max	0.219	0.005
WP avg	0.226	-0.025

Table 5: Correlations between $F_{0.1}$ and WordNet similarity metrics: path similarity (PS), Leacock Chodorow similarity (LCH), and Wu Palmer similarity (WP).

on noun similarity tasks, we hypothesize that this indicates better semantic structure for nouns than for verbs, further emphasizing that a possible limitation of the current baseline produced is its lack of precision when it comes to polysemous verbs. However, the positive correlation values seen for nouns, coupled with noun similarity performance, offer strong indications that the $F_{0.1}$ does provide useful semantic structure that can be further increased through better modeling.

6 Future work

Here, several directions are highlighted along which we see this work being extended.

Better models. The development of more accurate models for predicting definitions involved in the pair-wise relations will produce more interesting and useful networks, especially with the magnitude of examples of sense usage. Precision of verb relations seems to be a critical component of a better model.

Supervision. Relationship prediction is currently unsupervised. While it is an interesting task to model in this fashion, crowd sourcing the annotation of this data would be possible through services like Amazon Mechanical Turk. This would allow for the potential of exploring supervised models for predicting relationship links, particularly for relationships like synonymy and antonymy which are familiar concepts for a broad community of potential annotators.

WordNet semi-supervision. Another logical transformation of this task would be to use WordNet to inform the induction of a network in a semi-supervised fashion. There are many ways to go about this such as using statistics from WordNet to create a loss function, or using the structure of

WordNet as a base. As this work aimed to create a network solely from the data of Wiktionary, these ideas were not explored. However, using WordNet in this fashion is one of the directions of greatest interest for exploration in the future.

Sense usage examples. The examples present in Wiktionary have only begun to be used in this work. When examples are pulled, the source definition and lemma are linked. However, these examples have the potential to be linked to other senses and lemmas. This would an immense amount of structured, sense-usage data that could be used for many machine learning tasks.

Multilingual networks Wiktionary has been explored as a multilingual resource in previous works (de Melo and Weikum, 2009; Gurevych et al., 2012; Meyer and Gurevych, 2012b; Bond and Foster, 2013) largely due to the natural alignment across languages. Extending this approach to a multilingual setting could prove to be extremely useful for machine translation, and could allow low resource languages to benefit from alignment with other languages that have more annotations.

7 Conclusion

This paper introduced the idea of constructing a wordnet solely using the data from Wiktionary. Wiktionary is a powerful resource, featuring millions of pages that describe lemmas, their senses, example usages, and the relationships between them. Previous work has explored aligning resources like this with other networks like the Princeton WordNet. However, no work has fully explored the idea of building an entire network from the ground up using just Wiktionary.

This work explores simple baselines for constructing a network from Wiktionary through antonym and synonym relationships and compares induced networks with WordNet to find similar structures and statistics that appear to highlight strong future directions of particular interest, including but not limited to improving network modeling, linking more semantic examples, and reinforcing network construction using expert-annotated networks, like WordNet.

As conducted, this work is an initial step in transforming Wikitionary from an open-source dictionary into a powerful tool, dataset, and framework, with the hope of driving and motivating further work at endeavors studying languages and developing language processing systems.

References

Yoshua Bengio, Rjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3:1137–1155.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of machine learning research*, 3:993–1022.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Francis Bond and Ryan Foster. 2013. Linking and extending an open multilingual wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1352–1362, Sofia, Bulgaria. Association for Computational Linguistics.

Lars Borin, Jens Allwood, and Gerard de Melo. 2014. Bring vs. mtroget: Evaluating automatic thesaurus translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2115–2121, Reykjavik, Iceland. European Language Resources Association (ELRA).

Nello Cristianini and John Shawe-Taylor. 2000. *An introduction to Support Vector Machines: and other kernel-based learning methods*. Cambridge University Press, New York, NY, USA.

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

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. *ACM Trans. Inf. Syst.*, 20:116–131.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1606–1611, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Anne Göhring. 2014. Building a spanish-german dictionary for hybrid mt. In *Proceedings of the 3rd Workshop on Hybrid Approaches to Machine Translation (HyTra)*, pages 30–35, Gothenburg, Sweden. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Sanda M. Harabagiu, George A. Miller, and Dan I. Moldovan. 1999. Wordnet 2 - a morphologically and semantically enhanced resource. In *SIGLEX99: Standardizing Lexical Resources*.

Mohammad Javad Hosseini, Noah A. Smith, and Su-In Lee. 2016. Uw-cse at semeval-2016 task 10: Detecting multiword expressions and supersenses using double-chained conditional random fields. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 931–936, San Diego, California. Association for Computational Linguistics.

Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Machine Learning: ECML-98*, pages 137–142, Berlin, Heidelberg. Springer Berlin Heidelberg.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

David Jurgens and Mohammad Taher Pilehvar. 2015. Reserating the awesometastic: An automatic extension of the wordnet taxonomy for novel terms. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1459–1465, Denver, Colorado. Association for Computational Linguistics.

Robert Krovetz. 1992. Sense-linking in a machine readable dictionary. In *30th Annual Meeting of the Association for Computational Linguistics*, pages 330–332, Newark, Delaware, USA. Association for Computational Linguistics.

Thomas K. Landauer and Susan T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

C. Leacock and M. Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. *WordNet: An Electronic Lexical Database*, pages 265–283.

J. Lilleberg, Y. Zhu, and Y. Zhang. 2015. Support vector machines and word2vec for text classification with semantic features. In *2015 IEEE 14th International Conference on Cognitive Informatics Cognitive Computing (ICCI*CC)*, pages 136–140.

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

Michael Matuschek, Christian M. Meyer, and Iryna Gurevych. 2013. Multilingual knowledge in aligned wiktionary and omegawiki for translation applications. *Translation: Computation, Corpora, Cognition*, 3(1).

Gerard de Melo and Gerhard Weikum. 2009. Towards a universal wordnet by learning from combined evidence. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 513–522, New York, NY, USA. ACM.

Christian M. Meyer and Iryna Gurevych. 2010a. How web communities analyze human language: Word senses in wiktionary. *Proceedings of the 2nd Web Science Conference*.

Christian M. Meyer and Iryna Gurevych. 2010b. Worth its weight in gold or yet another resource–a comparative study of wiktionary, openthesaurus and germanet. In *Computational Linguistics and Intelligent Text Processing*, pages 38–49, Berlin, Heidelberg. Springer Berlin Heidelberg.

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

Christian M. Meyer and Iryna Gurevych. 2012a. Ontowiktionary — constructing an ontology from the collaborative online dictionary wiktionary. In M. T. Pazienza and A. Stellato, editors, *Semi-Automatic Ontology Development: Processes and Resources*, pages 131—161. IGI Global, Hershey, PA.

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 COLING 2012*, pages 1763–1780, Mumbai, India. The COLING 2012 Organizing Committee.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

George A. Miller. 1995. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–31.

George A. Miller and Walter G. Charles. 1991. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

George A. Miller and Florentina Hristea. 2006. Squibs and discussions: Wordnet nouns: Classes and instances. *Computational Linguistics*, 32(1):1–3.

Tristan Miller and Iryna Gurevych. 2014. Wordnet—wikipedia—wiktionary: Construction of a three-way alignment. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2094–2100, Reykjavik, Iceland. European Language Resources Association (ELRA).

Grace Muzny and Luke Zettlemoyer. 2013. Automatic idiom identification in Wiktionary. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1417–1421, Seattle, Washington, USA. Association for Computational Linguistics.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 113–120, Sydney, Australia. Association for Computational Linguistics.

Patrick Pantel and Marco Pennacchiotti. 2008. Automatically harvesting and ontologizing semantic relations. In *Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pages 171–198, Amsterdam, Netherlands. IOS Press.

Karl Pearson. 1895. Notes on regression and inheritance in the case of two parents. In *Proceedings of the Royal Society of London*, volume 58, pages 240–242. The Royal Society of London.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Herbert Rubenstein and John Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8:627–633.

Jon Rusert and Ted Pedersen. 2016. Umnduluth at semeval-2016 task 14: Wordnet's missing lemmas. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1346–1350, San Diego, California. Association for Computational Linguistics.

Bahar Salehi, Paul Cook, and Timothy Baldwin. 2014. Detecting non-compositional mwe components using wiktionary. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1792–1797, Doha, Qatar. Association for Computational Linguistics.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. pages 298–307.

Bernhard Schölkopf, Chris Burges, and Vladimir Vapnik. 1995. Extracting support data for a given task. In *Proceedings of the First International Conference on Knowledge Discovery and Data Mining*, KDD'95, pages 252–257. AAAI Press.

C Spearman. 2010. The proof and measurement of association between two things. *International Journal of Epidemiology*, 39(5):1137–1150.

Wikimedia. Wiktionary, the free dictionary.

Lingfei Wu, Ian En-Hsu Yen, Kun Xu, Fangli Xu, Avinash Balakrishnan, Pin-Yu Chen, Pradeep Ravikumar, and Michael J. Witbrock. 2018. Word mover's embedding: From word2vec to document embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4524–4534, Brussels, Belgium. Association for Computational Linguistics.

Z. Wu and M. Palmer. 1994. Verbs semantics and lexical selection. *Proceedings of the 32nd conference on Association for Computational Linguistics*, pages 133–138.

Dongqiang Yang and David Powers. 2006. Verb similarity on the taxonomy of wordnet. *Proceedings of the 3rd International WordNet Conference (GWC)*, pages 121–128.

Torsten Zesch, Christof Muller, and Iryna Gurevych. 2008. Using wiktionary for computing semantic relatedness. In *In Proceedings of AAAI*, pages 861–867.