Towards a Universal Wordnet by Learning from Combined Evidence

Gerard de Melo
Max Planck Institute for Informatics
Saarbrücken, Germany
demelo@mpi-inf.mpg.de

Gerhard Weikum
Max Planck Institute for Informatics
Saarbrücken, Germany
weikum@mpi-inf.mpg.de

ABSTRACT

Lexical databases are invaluable sources of knowledge about words and their meanings, with numerous applications in areas like NLP, IR, and AI. We propose a methodology for the automatic construction of a large-scale multilingual lexical database where words of many languages are hierarchically organized in terms of their meanings and their semantic relations to other words. This resource is bootstrapped from WordNet, a well-known English-language resource. Our approach extends WordNet with around 1.5 million meaning links for 800,000 words in over 200 languages, drawing on evidence extracted from a variety of resources including existing (monolingual) wordnets, (mostly bilingual) translation dictionaries, and parallel corpora. Graph-based scoring functions and statistical learning techniques are used to iteratively integrate this information and build an output graph. Experiments show that this wordnet has a high level of precision and coverage, and that it can be useful in applied tasks such as cross-lingual text classification.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods; I.2.6 [**Artificial Intelligence**]: Learning

General Terms

Algorithms

1. INTRODUCTION

Motivation. With the increasing degree of Internet penetration all over the world, the English language represents a constantly decreasing fraction of the Web. China and the EU each have greatly surpassed the U.S. in the number of Internet users, and other regions are expected to follow. Multilingual knowledge bases address this development by capturing relationships between words and concepts and hence making the semantic connections between words in different languages explicit. Lexical information of this sort can be useful for various forms of natural language processing [20], information retrieval (e.g. query expansion [18], cross-lingual IR [13], and question answering [33]), knowledge management (e.g. ontology construction [36] and ontology mapping [22]), artificial intelligence (e.g. textual entailment [3] and visual object recognition [27]), as well as human consultation. For example, knowing that the French words '*étudiant*', '*élève*', '*écolier*' are synonymous can aid in query expansion, and knowing that '*lycée*', '*école*', '*université*', '*académie*' are all specific types of educational institutions is helpful for question answering.

Contribution. In this paper, we present new methods for automatically creating a large-scale multilingual lexical database that organizes over 800,000 words from over 200 languages in a hierarchically structured semantic network, providing over 1.5 million links from words to word meanings. This universal wordnet (*UWN*) is bootstrapped from the Princeton WordNet, a well-known lexical database for the English language [14] that we shall simply refer to as '*WordNet*', in contrast to the generic term '*wordnet*'. WordNet consists of about 150,000 *terms* (words or short phrases) and about 120,000 *word senses* (concepts). It links terms with the senses that they denote (their meanings), thus providing a fairly comprehensive database of *synonymy* and *polysemy*. Additionally, it connects senses by semantic relationships like *hypernymy*, which is similar to the subclass relation and hence induces a hierarchical organization, as well as *meronymy* (part/whole relation), etc. For instance, '*high school*' is a hyponym of '*educational institution*', and '*classroom*' is a meronym (part) of '*schoolhouse*'. Similar *wordnets* do exist for about 50 different languages, but none of them are nearly as complete as the original English WordNet – many are small and unmaintained. Moreover, for many actively used languages, no such lexical databases exist at all. Our work addresses this gap and goes beyond the notion of monolingual wordnets by constructing an integrated multilingual wordnet that maps terms (words, phrases) of many languages to their meanings in the language-independent space of senses (concepts). This allows, for example, finding Greek hypernyms of the German word '*Schulgebäude*' ('*school building*'). This level of semantic connections and support for IR and AI tasks can never be reached by a mere translation dictionary between two languages.

Overview. Our method for building UWN starts with a limited number of existing (monolingual) wordnets to derive a large set of *senses*, i.e., possible word meanings, represented in a graph G_0 of terms and senses. This graph is extended by extracting information from a range of sources like (mostly bilingual) translation dictionaries, (monolingual) thesauri, and parallel corpora, as well as applying automatic procedures. Statistical methods are then used to link terms in different languages to adequate senses (the words' meanings) by analysing this graph, as illustrated in Figure 1. The left side depicts the input graph created from monolingual wordnets and translation dictionaries. The right side shows the output

CIKM'09, November 2–6, 2009, Hong Kong, China.
Copyright 2009 ACM 978-1-60558-512-3/09/11 ...$10.00.

graph where several words in different languages have been connected to the sense nodes that represent their possible meanings. The difficulty is determining which senses apply to which translations, e.g. a simple English word such as '*class*' has 9 senses listed in WordNet, '*form*' has 23 senses, and there are extreme examples such as the word '*break*', for which 75 different senses are enumerated. We attempt to discern disambiguation information in a series of graph refinements. To this end, we construct a rich set of numeric features for assessing the validity of a graph's edges. We train a support vector machine (SVM) over this feature space with a small number of hand-labelled edges. Then the SVM can automatically discriminate edges that are likely to be valid from spurious ones. The algorithm runs iteratively, i.e. several graphs G_i may be constructed, each refining the previous graph G_{i-1} by recomputing features and re-applying the SVM learner.

The rest of the paper is organized as follows. Section 2 reviews WordNet and related work. Section 3 describes the initial graph construction phase. Section 4 presents the feature space and learning model for graph refinement. Section 5 shows experimental results that confirm the high recall and precision of our method, and demonstrates the benefit for cross-lingual text classification.

2. WORDNET AND RELATED WORK

The original WordNet [14] was manually compiled at Princeton University to evaluate hypotheses about human cognition, but rapidly became one of the most widely used lexical resources for English natural language processing.

WordNet has sparked a number of endeavours aiming at similar databases for other languages, most importantly perhaps the EuroWordNet [40] and BalkaNet projects [39] that targeted many European languages. Individual institutes have made similar efforts for further languages, often under the auspices of the Global WordNet Association. Unfortunately, the work on such resources has not resulted in a unified multilingual wordnet, as there are different sense identifiers, formats, licences, etc.

Previous attempts to address this situation are still in their infancy. Marchetti et al. [26] proposed a Semantic Web tool for managing and interlinking wordnets in order to create a multilingual grid, however they do not focus on the problem of actually populating this grid. Another ambitious project started in 2006, the Global Wordnet Grid [15], only contains very limited sets of concepts for English, Spanish, and Catalan, as of August 2009.

A central problem in establishing wordnets is the laborious manual compilation process, which typically leads to insufficient coverage for practical applications. Several authors have attempted to automatically or semi-automatically construct a wordnet for a not yet covered language using existing wordnets [29, 2, 7, 16, 11, 21]. Our approach adopts some of the basic ideas of their work, but goes beyond simple heuristics by computing more sophisticated features that can account for very subtle differences between correct and incorrect terms-sense mappings. Prior approaches have not been able to produce both high coverage and high precision. Many of them experienced difficulties with polysemous terms and were applied to nouns only, while our technique works particularly well for commonly used polysemous terms. Isahara et al. [21] attempted to use multiple existing wordnets to combine information from multiple translation dictionaries, however with precision scores of 54% at best. None of these studies have explored the ideas of letting automatically established mappings for different languages reinforce each other or of exploiting evidence from multilingual translation graphs. Finally, none of the previous approaches have been applied to the task of building a large-scale multilingual wordnet.

There are not many alternative approaches to multilingual lexical databases. The PANGLOSS ontology [24] was built in the 1990s to facilitate machine translation. Interesting linking heuristics were used; however no learning techniques were employed, and the final coverage was limited to around 70,000 entities in two languages. Cook [6] created a semantic network that incorporates WordNet and links nouns in three languages to wordnet nodes based on simple heuristics as well as manual work. The heuristics yield high-quality results but apply to monosemous nouns only and hence fail to account for most commonly used words, as these tend to be polysemous. A much larger lexical resource has been presented by Etzioni et al. [13], who use translation dictionaries and two versions of Wiktionary to create a very large translation graph, which is then exploited for cross-lingual image search. Their central aim, however, is to derive a translation resource rather than constructing a semantic network with terms and senses equipped with additional relations like hypernymy, meronymy, etc. More recently, Adar et al. [1] have shown how to utilize the cross-linkage between Wikipedia articles in different languages for aligning Wikipedia infoboxes. This task is concerned with individual named entities (persons, organizations, etc.) only; it does not address general terminologies and term-sense mappings.

3. INITIAL GRAPH CONSTRUCTION

Lexical knowledge bases can be treated as labelled graphs. We consider weighted labelled multi-digraphs $G = (V, A, \ell, \Sigma_\ell)$ where:

- V is a set of nodes
- $A \subseteq V \times V \times \Sigma_\ell \times [0,1]$ is a set of weighted labelled arcs
- $\ell : V \cup A \rightarrow \Sigma_\ell$ is a labelling function that yields the respective label for a given node or arc
- Σ_ℓ is the labelling alphabet for nodes and arcs, i.e. the set of possible labels, defined as the union of the sets given below

Node labels are taken from the following sets:

a) $T \times L$: for *term nodes* representing words or expressions, where T is the set of NFC-normalized Unicode character strings [9], and L is the set of ISO 639-3 language identifiers

b) $S \times C$: for *sense nodes* representing meanings, where S is the set of sense identifiers provided by WordNet and C is the set of lexical categories (`noun`, `verb`, `adjective`, etc.)

Arc labels are taken from:

a) $\{\texttt{translation}\} \times C \times C$: for term-to-term arcs that connect term nodes to their translations into other languages, with source and target lexical categories in C (e.g. `noun`, `verb`, etc., or most commonly `unknown`)

b) $\{\texttt{meaning}\} \times \mathbb{N} \times \{0, 1\}$: for arcs representing links from terms to their meanings, with natural numbers representing sense frequencies or 1 if unavailable (see Section 3.1), as well as an indicator for distinguishing candidate arcs from imported arcs

c) $\{\texttt{lexicalization}\}$: for arcs that connect sense nodes back to their terms (the inverse of `meaning` arcs)

d) $\{\texttt{related}\}$: for term-to-term arcs that provide generic indications of semantic relatedness, e.g. between '*teach*' and '*university*'

e) $\{\texttt{hypernymy}\}$: for arcs between two sense nodes n_1, n_2 when n_2 denotes a generalization of the sense associated with n_1, e.g. n_1 could denote high schools and n_2 could denote educational institutions in general

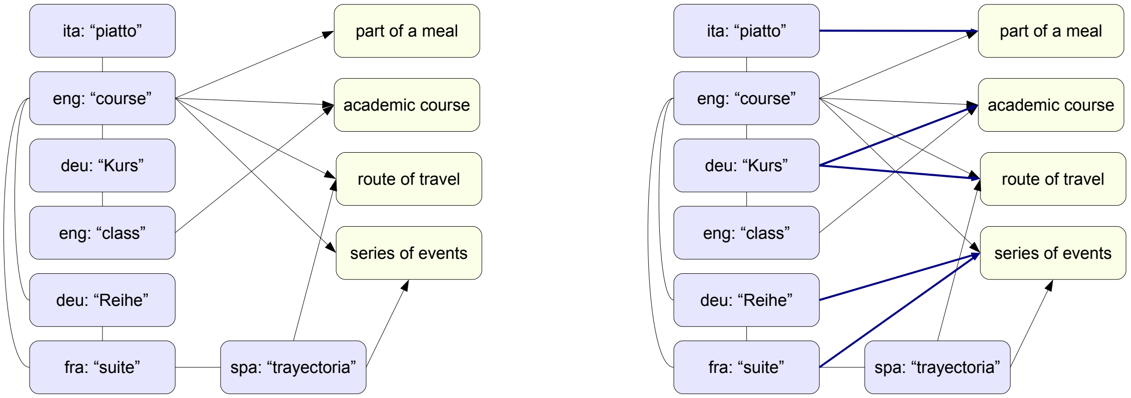

Figure 1: Arcs in the input graph G_0 (left) and the desired output graph G_i (right). Lines with arrows represent `meaning` arcs from term nodes to sense nodes, while each line without an arrow represents two reciprocal `translation` arcs.

.

f) $\{$`meronymy`, `antonymy`, $\dots\}$: for other lexico-semantic relationships provided by WordNet

For brevity, we shall use $\Gamma_{\mathrm{i}}(n, A) = \{n' \mid \exists l, w : (n', n, l, w) \in A\}$ to denote the in-neighbourhood, and $\Gamma_{\mathrm{o}}(n, A) = \{n' \mid \exists l, w : (n, n', l, w) \in A\}$ for the out-neighbourhood of a node, given a set of arcs A. In the following, we will work with multiple graphs of the form just described. The initial input graph G_0 will be the result of an extraction and synthesis of data from existing sources, while further graphs G_i ($i \geq 1$) constructed later on will extend G_0 with statistically derived information that eventually yield the multilingual UWN graph.

3.1 Information Extraction and Acquisition

The initial graph $G_0 = (V, A_0, \ell_0, \Sigma_\ell)$ is populated by extracting information from a range of different sources. Most imported arcs have a weight of 1, while a large number of so-called *candidate arcs* will be established with a weight of 0.

Existing Wordnet Instances. To bootstrap the construction, we rely on existing wordnets to provide term-to-sense `meaning` arcs for a limited set of languages, as well as sense-to-sense arcs (e.g. `hypernymy`) as described earlier. Apart from Princeton WordNet 3.0, such information is also taken from the Arabic, Catalan, Estonian, Hebrew, and Spanish wordnets[1], as well as from the human-verified parts of MLSN [6]. Edges have a weight of 1 except in some cases where mappings between different versions of WordNet [8] are applied to obtain uniform sense identifiers. Sense frequency information for the sense-annotated SemCor corpus [14] is incorporated as an annotation into `meaning` arc labels. Such information reveals to us how often for example the word ‘*school*’ was used to refer to a school building in the corpus.

Translation Dictionaries. A considerable number of term-to-term `translation` arcs with weight 1 are imported from over 100 open-source translation dictionaries that are freely available on the Web[2]. As only few such resources consist of well-structured XML, making their information amenable to machine processing frequently requires custom preprocessing steps. These involve separating the actual terms from annotation information such as part-of-speech (e.g. `adverb`), semantic domain (e.g. `chemistry`), etc. We treat translation information as $n : n$ relationships between words, adding source or target part-of-speech labels to the translation arcs whenever they are given.

Wiktionary. The community-maintained Wiktionary project[3] offers a plethora of lexical information but relies on simple text-based mark-up rather than an explicit, precise database schema. We thus use rule-based information extraction techniques to mine `translation` and other arcs from eight different language versions of Wiktionary.

Multilingual Thesauri and Ontologies. Translations are also obtained from concept-oriented resources such as the GEneral Multilingual Environmental Thesaurus (GEMET[4]), OmegaWiki[5], as well as from OWL ontologies [4]. For each sense (concept) C, we consider its set of label terms $T(C)$ in the resource, and then add a `translation` arc to the graph for each $t_i, t_j \in T(C)$ ($i \neq j$), unless they are from the same language, in which case we create a `related` arc instead.

Parallel Corpora. Text from conventional multilingual corpora, translation memories, film subtitles, and software localization files can be word-aligned to harness additional translation information for many language pairs. We make use of GIZA++ [28] and Uplug [37] to produce lexical alignments for a subset of the OPUS corpora [38], which includes the OpenSubtitles corpus. Since word alignments tend to be unreliable, we compile alignment statistics and add `translation` arcs to the graph between pairs of nodes where the respective term pair is encountered with a high frequency (above a specified threshold).

Monolingual Thesauri. Monolingual thesauri from the OpenOffice software distribution[6] provide `related` arcs between the terms of a single language, revealing e.g. that ‘*college*’ is semantically related to ‘*university*’.

Manually Classified Arcs. As our approach is based on supervised learning, we also depend on a limited amount of manually classified `meaning` arcs from terms to senses, obtained via a collaborative Web editing environment. Such arcs are either labelled as positive (correct, adequate) or negative (incorrect, inadequate).

[1] See http://www.globalwordnet.org/

[2] For example from the FreeDict project, http://www.freedict.org

[3] http://www.wiktionary.org

[4] http://www.eionet.europa.eu/gemet/

[5] http://www.omegawiki.org

[6] http://wiki.services.openoffice.org/wiki/Dictionaries

3.2 Graph Enrichment and Pruning

After the initial information extraction, we apply additional preprocessing heuristics to the input graph.

First of all, we assume the `translation` relation is symmetric and add inverse translation arcs to ensure such links are reciprocal. Additionally, while the relation is not transitive, we use so-called triangulation heuristics to reduce the sparsity of translations. For instance, when the Italian word '*scuola*' has an English translation '*school*' and a French translation '*école*', and the latter two both have a Malay translation '*sekolah*', then we can infer that this Malay word is also a likely translation for the Italian term. Translation arcs between two term nodes n_1, n_2 are added when

$$|\{n'|n' \in \Gamma_{\mathrm{o}}(n_1, A_0) \cap \Gamma_{\mathrm{i}}(n_2, A_0)\}| \geq m_{\min}$$

where $m_{\min} = 5$ was chosen empirically for high accuracy.

Subsequently, the graph is pruned by merging duplicate and near-duplicate arcs as follows. We define a partial ordering $\leq_\ell$ over arc labels that captures when an arc label is considered more specific than another one. We assume that specific labels should be preferred over generic ones, e.g. when we have two translation arcs, one without and one with lexical category information, we choose to keep only the latter although the translation might also hold for other lexical categories. This then allows us to iterate over all arcs $a = (n_1, n_2, l, w) \in A_0$, discarding a whenever there exists another arc $a' = (n_1, n_2, l', w')$ with $l \leq_\ell l'$, $w \leq w'$.

3.3 Candidate Arc Creation

As a final preprocessing step that concludes the construction of G_0, we create a large set of zero-weighted arcs that denote *potential* relationships between words and meanings that will later be evaluated. As candidate meanings we consider all senses of translations of a given term. We determine all 2-hop paths of the form $\{(n_0, n_1, l, w), (n_1, n_2, l', w')\} \subset A_0$, where n_0 is a term node, the arc label l is a `translation` one, and n_2 is a sense node. For each such path, a new *candidate arc* $(n_0, n_2, l_{\mathrm{m}}, 0)$ is created, linking a term to one of its potential senses, where l_{m} identifies the arcs as zero-weighted `meaning` ones and as candidates. For instance, in Figure 1, the Italian word '*piatto*' has a translation arc to '*course*', which in turn has four outgoing meaning arcs in G_0, so four candidate arcs will be created for '*piatto*'. The arc to the sense described as '*part of a meal*' would be an adequate candidate arc for '*piatto*' that should later receive a higher weight, while the other senses, e.g. '*academic course*' are inadequate, and should no longer be present in the final output graph.

4. ITERATIVE GRAPH REFINEMENT

In each iteration, a new graph $G_i = (V, A_i, \ell_i, \Sigma_\ell)$ is constructed that is topologically identical to G_{i-1} and thus to G_0. However, the weights of all candidate `meaning` arcs are re-assessed to reflect a refined measure of confidence in them being correct. To this end, our approach is to learn a statistical model for assessing the validity of candidate arcs. We employ a supervised classifier that is trained by the small set of hand-labelled arcs included in G_0, which are labelled either as correct (positive training samples) or incorrect (negative training samples). For a given candidate arc, it predicts a weight in $[0, 1]$ that represents the degree of confidence in the respective arc being correct, given the previous graph G_{i-1}.

The classifier operates over an appropriately defined feature space. In our approach, the feature space is recomputed with each new graph G_i of the refinement process. This is in the spirit of relaxation labelling methods and belief propagation methods for graphical models [17]. Directly applying standard relational learning algorithms to the huge graph in our task would face tremendous scalability problems, since we need to capture certain non-straightforward dependencies between different arcs and nodes even when they are several hops apart. Therefore, we embed information about the neighbourhood of an arc into its feature vector. In the ideal case, the weight of an arc, given its feature vector, will then be conditionally independent of the weights of other arcs, allowing us to use a more traditional learner. In each iteration i, the previous graph G_{i-1} is used as the basis to derive a feature vector $\mathbf{x} \in \mathbb{R}^m$ for each candidate arc in G_i (where m is the number of features). Details will be given in Section 4.1.

Using the feature vectors for the hand-labelled training set, we train an RBF-kernel SVM classifier. SVMs are based on the idea of computing a separating hyperplane that maximizes the margin between positive and negative training instances in the feature space or in a high-dimensional kernel space [12]. For each feature vector $\mathbf{x}$, SVM classification yields values $f(\mathbf{x}) \in \mathbb{R}$, which correspond to distances from the separating hyperplane in the kernel space. To obtain arc weights, we adopt Platt's method of estimating posterior probabilities using a sigmoid function $w := P(w = 1|\mathbf{x}) = \frac{1}{1+\exp(af(\mathbf{x})+b)}$, where parameter fitting for a and b is performed using maximum likelihood estimation on the training data [30, 25]. This allows us to obtain new arc weights in $w \in [0, 1]$ for all candidate arcs from term to sense nodes, concluding the construction of the new graph G_i.

4.1 Feature Computation

For each candidate `meaning` arc (n_0, n_2, l, w) in G_i, we quantify evidence from the graph as an m-tuple of numerical scores $\mathbf{x} = (x_1(n_0, n_2), \ldots, x_m(n_0, n_2)) \in \mathbb{R}^m$, such that the learning algorithm be able to assess whether the arc should be accepted. We expect to see strong evidence for this arc if n_2, a sense node, denotes one of the senses of n_0, a term node. Given the previous graph G_{i-1}, we compute scores $x_i(n_0, n_2)$ as listed in Table 1. In Equation 1, two nodes are directly compared by means of a cosine-based context similarity score, which will be explained in Subsection 4.1.3. The underlying idea for Equations 2 and 3 (where ϕ_1, ϕ_2, γ are arc and path weighting functions) is that a word's most likely senses can be determined by considering likely senses n_2' of its translations and related terms $n_1 \in \Gamma_{\mathrm{o}}(n_0, A_{i-1})$. Equation 2 considers each successor node n_1, and looks at how similar the successors of n_1 are to n_2. For instance, in the simplest case, if we use an identity test as a similarity function for comparing those successors n_2' to n_2, then this score effectively computes a weighted count of the number of two-hop paths from n_0 to n_2. For example, in Figure 1, there are multiple paths from the German word '*Kurs*' to the '*academic course*' sense node. Equation 3 is similar, but normalizes with respect to the number of alternative choices in the denominator. In the simplest case, the dissim function will simply count how many alternative senses there are, so if n_1 has n_2 as one of its senses, and 4 other senses, it would return 4, and lead to a summand of $\frac{1}{1+4}$ for n_1, which reflects the probability of reaching n_2 from n_1. Equation 3 is also applied in the opposite direction to quantify reachability information from a sense node to a term node.

More sophisticated scores are obtained by applying additional weighting and normalization. The scores depend on a number of auxiliary formulae, in particular combinations of arc weighting functions ϕ_1, ϕ_2, as described in Section 4.1.1, path weighting functions γ, described in Section 4.1.2, and measures of semantic relatedness, described in Section 4.1.3. For example, in Equation 3 we may wish to not count all alternative senses, instead producing a weighted score where alternative senses are not fully considered if they are very similar or if their lexical category tags do not match.

4.1.1 Arc Weighting Functions

The different versions of ϕ_1 listed in Table 2 estimate the relevance of a connection from a term n_0 to a translation or related term n_1. Equation 6 filters out `related` arcs, while Equation 7 considers the size of the out-neighbourhood of n_0, counting the number of terms that have outgoing `meaning` arcs. Equation 8 is similar to Equation 3 and normalizes with respect to a weighted in-degree of n_1 for terms from the same language.

Instantiations of ϕ_2 estimate the relevance of connections from translations or related terms n_1 to sense nodes n_2. For this, Equation 10 considers the weights of `meaning` arcs, while Equation 11 uses sense corpus frequencies. Equations 9 and 12 are helper functions.

4.1.2 Cross-Lingual Lexical Category Heuristics

Several features described in Table 1 integrate a function γ that assigns weights to paths in the graph. Apart from the trivial choice of setting it to a constant value, we use γ^{lc} as a version that considers lexical categories (part-of-speech tags) associated with nodes in the graph. Many of the previous studies on automatically building wordnets dealt with nouns exclusively, whereas all lexical categories are respected in our approach, so some means of preventing, for example, a noun from being mapped to a verb sense is required.

$\gamma^{lc}(n_0, \dots, n_k)$ is supposed to estimate whether the nodes along the path from n_0 to n_k have the same or at least compatible lexical categories. It is computed as

$$\gamma^{lc}(n_0, \dots, n_k) = \max_{c \in C} \prod_{i=1}^{k-1} \mu_c(n_i, n_{i+1}).$$

Here, $\mu_c(n_i, n_{i+1})$ estimates whether a local transition from n_i to n_{i+1} is possible with category $c \in C$ with the following heuristics.

1. In some cases, there may be a `translation` arc with matching lexical categories. As explained earlier in Sections 3 and 3.1, some dictionaries provide part-of-speech information that is extracted and included as part of the arc's label.
2. When this fails, we compare *possible* categories of n_i and n_{i+1}. Categories for sense nodes can be derived from their node labels. For term nodes, we first check if the term has *any* incoming or outgoing `translation` arc labelled with c, or *any* `meaning` arc to a sense node labelled with c.
3. If this fails, we attempt to use learnt models for surface properties of term strings, which often reveal likely lexical categories. For each lexical category and language, we check whether criterion 3 above provides us with sufficient examples to create a training set and a withheld validation set (disjoint from the training set) of part-of-speech labelled terms. If so, we learn surface form properties as described below.
4. If none of the aforementioned steps apply, a default score of 0.5 may be used, which means that we assume the chance of a compatible lexical category to be 50%.

The surface form learning is carried out by growing a C4.5 decision tree [12] with the following features:

1. Prefixes and suffixes of a word up to a length of 10 (without case conversion): In many languages, affixes mark the part-of-speech tag of a word. For instance, in Italian, lemma forms of virtually all verbs end in '*-are*', '*-ere*', or '*-ire*'.
2. Boolean features for first character capitalization and complete capitalization: In many languages, capitalized words tend to be nouns (e.g. acronyms such as '*USA*', proper nouns like '*London*', all nouns in German, Luxemburgish).

The reliability of the decision tree depends largely on the language. For each lexical category and language, we evaluated on the respective validation set, obtaining F_1-scores between 0.03 and 0.99. Later on, for a given term to be analysed, we use the confidence estimate c from the decision tree's leaves only in the following cases:

1. the F_1-score on the validation set was high
2. $c > 0.5$ and the precision on the validation set was high
3. $c < 0.5$ and the recall on the validation set was high

4.1.3 Measures of Semantic Relatedness

The feature vector computation also uses a set of different semantic relatedness measures. To see the potential benefit of this technique, consider the following example. The single sense of '*schoolhouse*' is related to the educational institution sense of the word '*school*', but not to the sense of '*school*' that refers to groups of fish. So, if a term node has translation arcs to both '*school*' and '*schoolhouse*', their semantic relatedness tells us that the educational senses of '*school*' are much more likely to be correct than the one referring to fish. We consider four different measures of semantic relatedness.

- $\mathrm{sim}_{\mathrm{id}}(n_a, n_b)$ is the identity indicator function, i.e. yields 1 if $n_a = n_b$, and 0 otherwise.
- $\mathrm{sim}_{\mathrm{n}}(n_a, n_b)$ considers the graph neighbourhood. For a given path in the graph, we compute a proximity score multiplicatively from relation-specific arc weights (e.g. 0.8 for hypernymy, 0.7 for holonymy). The similarity is then defined to be the maximum score for all paths between n_a and n_b if this maximum is above or equal a pre-defined threshold $\alpha_n = 0.35$, and 0 otherwise. It can be obtained efficiently using a variant of Dijkstra's shortest-path algorithm [10].
- $\mathrm{sim}_{\mathrm{c}}(n_a, n_b)$ uses the cosine similarity of context strings for nodes. For senses, context strings are constructed by concatenating English sense descriptions (WordNet glosses) and terms linked to the original sense and neighbouring senses. For terms, the set of all English translations is used. Two context strings are compared by stemming using Porter's method, creating TF-IDF vectors $\mathbf{x}_a, \mathbf{x}_b$, and computing the cosine of the angle between them, i.e. $\mathbf{x}_a^T \mathbf{x}_b (||\mathbf{x}_a|| \, ||\mathbf{x}_b||)^{-1}$.
- $\mathrm{sim}_{\mathrm{m}}(n_a, n_b) = \max\{\mathrm{sim}_{\mathrm{n}}(n_a, n_b), \mathrm{sim}_{\mathrm{c}}(n_a, n_b)\}$ combines the power of $\mathrm{sim}_{\mathrm{n}}$, and $\mathrm{sim}_{\mathrm{c}}$, which are each based on rather different characteristics of the senses.

4.2 Iterative Procedure

Our iterative learning procedure makes use not only of the small set of manually classified `meaning` arcs supplied as training instances, but also benefits from the enormous numbers of originally unlabelled instances. There is often some form of mutual reinforcement of correct and highly weighted (but not known to be correct) arcs and there is some gradual down-weighting of incorrect arcs in the course of the iterations. Thus, our method can be seen as a form of semi-supervised learning. As a stopping criterion, we use either a withheld validation set of manually classified arcs (not used for training) or apply cross-validation with the training data, and check if a loss function $L(G_i)$ shows a reduction $L(G_{i-1}) - L(G_i) \geq \epsilon$ (where $epsilon$ may also be slightly negative). In practice, we observed that 2-4 iterations suffice to stabilize the precision and recall measures on the graph.

Having determined the most profitable iteration $i^* = \arg\max_i L'(G_i)$ with a loss function L' (possibly different from L), we can transform G_{i^*} into the final UWN graph G'_{i^*} using the following steps:

Table 1: Feature computation formulae, where $\mathrm{sim}^*_{n_0,\phi_2}(n_1,n_2)$ yields the maximum weighted similarity between successors of n_1 and n_2, and $\mathrm{dissim}_{n_0,\phi_2}(n_1,n_2)$ produces weighted sums of dissimilarities between successors of n_1 and n_2

$$x_i(n_0,n_2) = \mathrm{sim}(n_0,n_2) \tag{1}$$

$$x_i(n_0,n_2) = \sum_{n_1\in\Gamma_{\mathrm{o}}(n_0,A_{i-1})} \phi_1(n_0,n_1)\ \mathrm{sim}^*_{n_0,\phi_2}(n_1,n_2) \tag{2}$$

$$x_i(n_0,n_2) = \sum_{n_1\in\Gamma_{\mathrm{o}}(n_0,A_{i-1})} \phi_1(n_0,n_1)\ \frac{\mathrm{sim}^*_{n_0,\phi_2}(n_1,n_2)}{\mathrm{sim}^*_{n_0,\phi_2}(n_1,n_2)\ +\mathrm{dissim}_{n_0,\phi_2}(n_1,n_2)} \tag{3}$$

$$\mathrm{sim}^*_{n_0,\phi_2}(n_1,n_2) = \max_{n_2'\in\Gamma_{\mathrm{o}}(n_1,A_{i-1})} \gamma(n_0,n_1,n_2')\ \phi_2(n_1,n_2')\ \mathrm{sim}(n_2,n_2') \tag{4}$$

$$\mathrm{dissim}_{n_0,\phi_2}(n_1,n_2) = \sum_{n_2'\in\Gamma_{\mathrm{o}}(n_1,A_{i-1})} \gamma(n_0,n_1,n_2')\ \phi_2(n_1,n_2')(1-\mathrm{sim}(n_2,n_2')) \tag{5}$$

Table 2: Arc weighting functions plugged into the formulae in Table 1, where ϕ_3^{slc} compares the part-of-speech of sense nodes, $\mathrm{freq}(n_1,n_2)$ yields the frequency of term n_1 with sense n_2 in the SemCor corpus (or 1 if n_1 does not occur in the corpus), and N_s is the set of all sense nodes in the graph. ℓ_{i-1} is the labelling function of G_{i-1}, which, among other things, captures languages and lexical categories.

$$\text{Filtering}\quad \phi_1^{\mathrm{f}}(n_0,n_1) = \begin{cases}1 & \exists(n_0,n_1,l,w)\in A_{i-1} : l \neq \texttt{related}\\ 0 & \text{otherwise}\end{cases} \tag{6}$$

$$\text{Normalization}\quad \phi_1^{\mathrm{nm}}(n_0,n_1) = \frac{1}{|\{n_1\in\Gamma_{\mathrm{o}}(n_0,A_{i-1})\mid \Gamma_{\mathrm{o}}(n_1,A_{i-1})\cap N_s\neq\emptyset\}|} \tag{7}$$

$$\text{Weighted In-Degree}\quad \phi_1^{\mathrm{bt}}(n_0,n_1) = \frac{\mathrm{sim}^*_{n_0,\phi_1^{\mathrm{ln}}}(n_1,n_0)}{\mathrm{sim}^*_{n_0,\phi_1^{\mathrm{ln}}}(n_1,n_0)+\mathrm{dissim}_{n_0,\phi_1^{\mathrm{ln}}}(n_1,n_0)} \tag{8}$$

$$\text{Language Matching}\quad \phi_1^{\mathrm{ln}}(n_0,n_0') = \begin{cases}1 & \ell_{i-1}(n_0),\ell_{i-1}(n_0')\ \text{provide same language code}\\ 0 & \text{otherwise}\end{cases} \tag{9}$$

$$\text{Thresholding}\quad \phi_2^{\mathrm{t}\alpha}(n_1,n_2) = \begin{cases}1 & \exists(n_1,n_2,l,w)\in A_{i-1} : w>\alpha\\ 0 & \text{otherwise}\end{cases} \tag{10}$$

$$\text{Corpus Freq.}\quad \phi_2^{\mathrm{cf}}(n_1,n_2) = \frac{\mathrm{freq}(n_1,n_2)}{\sum\limits_{n_2'\in\Gamma_{\mathrm{o}}(n_1,A_{i-1})} \phi_3^{\mathrm{slc}}(n_2,n_2')\ \mathrm{freq}(n_1,n_2')} \tag{11}$$

$$\text{Sense Lexical Category}\quad \phi_3^{\mathrm{slc}}(n_2,n_2') = \begin{cases}1 & \ell_{i-1}(n_2),\ell_{i-1}(n_2')\ \text{provide same lexical category}\\ 0 & \text{otherwise}\end{cases} \tag{12}$$

(i) We retain from G_{i^*} only arcs whose labels designate them as candidate `meaning` arcs or as from a specific set of language-independent sense-to-sense arcs from Princeton WordNet.

(ii) Optionally, we threshold using two parameters $w_{\min}$,$\hat{w}_{\min}$, retaining only arcs with either $w > w_{\min}$, or $w > \hat{w}_{\min}$ and $\neg\exists n_2', l', w' : (n_0, n_2', l', w') \in A, w' > w$. This enforces a minimal weight $w_{\min}$ or possibly a slightly lower weight $\hat{w}_{\min}$ in the absence of alternative arcs for n_0.

(iii) Finally, we remove all nodes of degree 0.

Omitting step (ii) leads to a statistical form of lexical database where edge weights provide the degree of confidence of a link. Weighted edges can be useful in certain application settings. Including this step yields a more conventional, unweighted lexical database where only high quality links are retained. Our specific choices of loss functions and thresholds are given in the section on experimental results.

5. EXPERIMENTAL EVALUATION

5.1 Implementation and Setup

We used the Java programming language to develop a platform-independent knowledge base processing framework. For efficiency reasons, the weighted labelled multi-digraphs were stored in custom binary format databases, with optimized index and data caching as well as Bloom filtering for reduced disk access, i.e. avoiding unnecessary reads when no data is available. This framework allowed us to flexibly plug together information extraction modules, knowledge base processors, as well as exporters and analysis modules into knowledge base processing pipelines. Our graph refinement procedure is integrated as a link processor that assesses links between two entities and produces new weights. For statistical learning, it relies on the LIBSVM implementation [5] using an RBF kernel $K(x,y) = \exp(-\frac{1}{m}||x-y||^2)$ where m is the number of features.

Table 3: Iterations of algorithm with validation set scores (for $w_{\min} = 0.7$, $\hat{w}_{\min} = 0.6$)

Graph	Precision	Recall	F_1	# accepted meaning arcs
G_0	N/A	0.00%	0.00%	0
G_1	83.96%	67.42%	74.79%	1,540,206
G_2	83.70%	68.48%	75.33%	1,594,652
G_3	83.89%	68.64%	75.50%	1,595,763
G_4	83.90%	67.88%	75.04%	1,573,395

Table 4: Precision of UWN Result Graph

Dataset	Sample Size	Precision (Wilson)
French	311	89.23% ± 3.39%
German	321	85.86% ± 3.76%
Mandarin Chinese	300	90.48% ± 3.26%

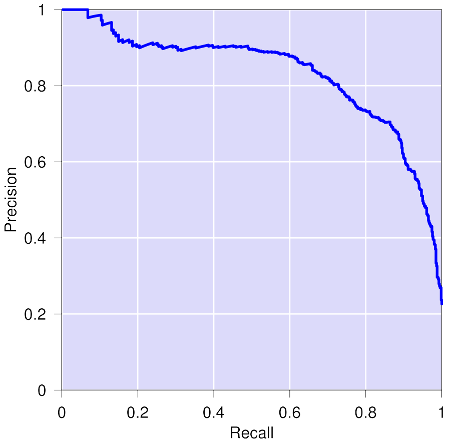

Figure 2: Precision-Recall curve on validation set for G_3 when $w_{\min} = \hat{w}_{\min}$

Following Section 3, G_0 was constructed with 448,069 existing `meaning` arcs (from the input wordnets, mainly English, Spanish, Catalan), 10,805,400 `translation` arcs (from the dictionaries, Wiktionary, thesauri and parallel corpora), and 10,343,601 candidate `meaning` arcs (generated following Section 3.3, on average 7.7 per term node). It contained roughly 129,500 sense nodes and 1.3 million term nodes with candidate arcs (5 million overall). We added 2,445 human-classified `meaning` arcs for training, out of which 610 were positive examples. The training set was compiled by manual annotation of candidate `meaning` arcs as either positive or negative for randomly selected French and German terms, rather than for randomly selected instances. This means that the risk of overfitting is reduced and the learner is channelled to focus explicitly on the distinction between negative and positive examples for a given word rather than coincidental differences between different words. We used a validation set of 2,901 French/German candidate `meaning` arcs, classified manually as positive or negative using the same methodology, and selected F_1 scores for this validation set on the output graph for $w_{\min} = 0.6$, $\hat{w}_{\min} = 0.5$ as the loss function.

5.2 Results for Meaning Arcs

The algorithm ran for four iterations until it failed to improve the F_1-score, as shown in Table 3. The input graph G_0 does not cover any of the validation arcs, and thus has a recall and F_1-score of 0%. English is the most widely represented language within the input graph, both with respect to the input wordnets and for the translations, so the first iteration provided for the most significant gains and already delivered excellent results. In the next iteration, G_1 served as the input graph, leading to an improved F_1-score for G_2 because a larger range of terms are equipped with non-zero `meaning` arcs in G_1 compared to G_0. These improvements decrease very quickly, since the additional amount of information available to the feature computation process, compared to previous iterations, keeps diminishing.

At this point, we have the choice of preferring high precision, e.g. G_2 has 91.59% precision at 44.55% recall for $w_{\min} = 0.9$, $\hat{w}_{\min} = 0.75$, or high recall, e.g. G_3 gives us 73.92% precision at 80.30% recall for $w_{\min} = 0.3$, $\hat{w}_{\min} = 0.25$. Our loss function balances precision and recall, making G_3 the most profitable graph. Figure 2 shows the tradeoff between precision and recall on G_3. For the final UWN output graph, we chose $w_{\min} = 0.6$, $\hat{w}_{\min} = 0.5$ as it provided good coverage at a reasonable precision. Figure 3 provides an excerpt from this graph, highlighting how words in different languages have been disambiguated to link to the appropriate senses of the English word '*school*', e.g., in French, the term '*banc*' is used to refer to a school of fish. We recruited human annotators for three languages, and asked them to evaluate randomly chosen arcs in the respective language from this output graph, relying on Wilson score intervals to generalize our findings in a statistically significant manner, as listed in Table 4. These randomly chosen arcs are not related to the training or validation sets, which moreover did not contain any Mandarin Chinese terms, so the results show that our learning approach applies cross-lingually. It must be pointed out that it is not possible to reliably evaluate the accuracy of a wordnet using pre-existing wordnets, as they do not fulfil the closed world assumption, i.e. a term-sense arc not occurring in an existing wordnet does not warrant the conclusion that the link is false. This is particularly true for current non-English wordnets, which often have limited coverage and sense inventories based on older versions of WordNet.

Table 5 shows the coverage of the output graph. Keeping in mind that the final UWN graph retains only *candidate* meaning arcs, these figures do not include any meaning arcs imported from the input wordnets, and only count term nodes that are connected to sense nodes via these new candidate meaning arcs. There are terms in more than 200 languages in UWN. The most well-represented languages result quite directly from the selection of translations in the

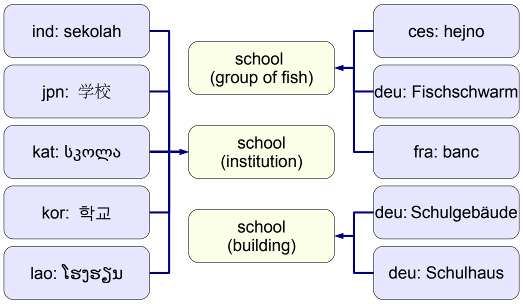

Figure 3: Excerpt from UWN graph with meaning arcs from terms to three sense nodes

Table 5: Coverage of final UWN graph with respect to accepted candidate `meaning` arcs as well as terms.

Language	Meaning Arcs	Distinct Terms
Overall	1,595,763	822,212
By Language		
German	132,523	67,087
French	75,544	33,423
Esperanto	71,247	33,664
Dutch	68,792	30,154
Spanish	68,445	32,143
Turkish	67,641	31,553
Czech	59,268	33,067
Russian	57,929	26,293
Portuguese	55,569	23,499
Italian	52,008	24,974
Hungarian	46,492	28,324
Thai	44,523	30,815
Others	795,782	427,216
By Lexical Category		
Nouns	1,048,003	589,536
Verbs	221,916	88,189
Adjectives	289,328	147,257
Adverbs	36,095	26,254

input graph G_0. We found that terms with translations to many languages had high chances of being included. Our approach thus addresses a long-standing problem in automatic construction of wordnets, namely that of insufficient coverage of commonly used words, which tend to be more polysemous. Using sophisticated features, it carefully benefits from cross-lingual evidence to find meanings of such terms, while previous approaches had trouble coping with the polysemy of commonly used words. The break-down by part-of-speech shows that the majority of terms are nouns. The terms in UWN have `meaning` links to a total of 80,620 distinct sense nodes. Table 6 shows average degrees with respect to `meaning` arcs for term nodes (out-degree) and sense nodes (in-degree), revealing the level of polysemy of terms according to UWN. The middle column shows average out-degrees when term nodes with only one `meaning` arc are excluded.

5.3 Results for Semantic Relations

We further evaluated to what extent relationships imported from Princeton WordNet apply to UWN. The intuition is that relations between senses, e.g. `hypernymy`, apply independently of the language of the terms associated with the respective senses. For several types of relations, at least 100 randomly selected links between two senses were assessed, where both senses have associated German language terms (linked via `meaning` arcs). Table 7 shows that the overall precision is high. Incorrect relationships resulted almost entirely from incorrect `meaning` arcs.

In addition to relations between senses, WordNet also provides relations between specific words with respect to senses of those words. Such relations cannot be applied directly to UWN, however, in some cases, we can infer from them more generic relationships between senses. For instance, when WordNet tells us that the word '*scholastic*' is derivationally related to the word '*school*', we can interpret this as a generic indicator of semantic relatedness. Antonymy relationships between words such as between '*good*' and '*bad*' are re-interpreted as a generic form of semantic opposition between senses. These, too, were evaluated in Table 7.

Table 6: Average degree with respect to meaning arcs of term nodes (out-degree) and sense nodes (in-degree)

	Term Node Out-Degree	Term Node Out-Degree Excluding Monosemous	Sense Node In-Degree (Multilingual)
Nouns	1.78	3.20	12.76
Verbs	2.52	4.24	16.12
Adjectives	1.96	3.63	15.19
Adverbs	1.37	2.53	9.97
Total	1.94	3.38	13.56

Table 7: Quality assessment for imported relations

Relation	Precision (Wilson interval)
hypernymy	87.1% ± 4.8%
instance	89.3% ± 4.4%
similarity	92.0% ± 3.8%
category	93.3% ± 4.5%
meronymy (part-of)	94.4% ± 4.1%
meronymy (member-of)	92.7% ± 4.0%
meronymy (substance-of)	95.6% ± 3.5%
antonymy (as sense opposition)	94.3% ± 3.9%
derivation (as semantic similarity)	94.5% ± 4.0%

5.4 Semantic Relatedness

We studied semantic relatedness assessment as an application of UWN in conjunction with Princeton WordNet's sense relations and descriptions. The objective is to automatically estimate the degree of relatedness between two words, producing scores that correlate well with the average ratings by human evaluators. For instance, '*curriculum*' is much more closely related to a word like '*school*' than '*water*'. Given two term nodes t_1, t_2, we estimate their relatedness as

$$\mathrm{rel}(t_1, t_2) = \max_{s_1 \in \Gamma_o(t_1, A)} \max_{s_2 \in \Gamma_o(t_2, A)} w(t_1, s_1) w(t_2, s_2) \mathrm{sim}(s_1, s_2)$$

using semantic relatedness measures for sense nodes described in Section 4.1.3 and $w(t, s)$ denoting the `meaning` arc weight. Three German-language datasets are compared with state-of-the-art scores obtained for GermaNet, the manually compiled German wordnet, and Wikipedia, as reported by Gurevych et al. [19]. In Table 8, the first row lists the inter-annotator agreement between different human evaluators and the number of term pairs rated for each dataset. The following rows show that UWN can be more useful than handcrafted resources, with respect to both the correlation with human judgments (Pearson product-moment correlation coefficient) and the coverage (the number of term pairs from the dataset where both terms are found in the respective lexical database).

5.5 Cross-Lingual Text Classification

Another applied task we considered was cross-lingual text classification. This is a very challenging task, where text documents are supposed to be classified, usually by topic, given only class-labelled training documents for a completely different language.

We preprocess a document by removing stop words and performing part-of-speech tagging as well as lemmatization using the TreeTagger [34]. In addition to the original term frequencies, we map each term to the respective sense nodes listed by UWN or by Princeton WordNet (for English words), embracing a rather sim-

Table 8: Evaluation of semantic relatedness measures, using Pearson's sample correlation coefficient. We apply our three semantic relatedness measures on the UWN graph and compare with the agreement between human annotators as well as scores for two alternative measures as reported by Gurevych et al. [19], one based on Wikipedia, the other on GermaNet.

Dataset	GUR65		GUR350		ZG222	
	Pearson r	Coverage	Pearson r	Coverage	Pearson r	Coverage
Inter-Annotator Agreement	0.81	(65)	0.69	(350)	0.49	(222)
Wikipedia (ESA)	0.56	65	0.52	333	0.32	205
GermaNet (Lin)	0.73	60	0.50	208	0.08	88
UWN (sim_n)	0.77	60	0.62	242	0.43	106
UWN (sim_c)	0.77	60	0.68	242	0.52	106
UWN (sim_m)	0.80	60	0.68	242	0.51	106

Table 9: Cross-lingual text classification in terms of micro-averaged precision, recall, and F_1-score.

	Precision	Recall	F_1
English-Italian			
Terms only	69.90%	66.81%	68.32%
Terms and senses	83.24%	70.49%	76.34%
English-Russian			
Terms only	57.86%	46.67%	51.66%
Terms and senses	67.87%	74.94%	71.23%
Italian-English			
Terms only	71.97%	77.06%	74.43%
Terms and senses	76.59%	79.67%	78.10%
Italian-Russian			
Terms only	59.65%	57.15%	58.37%
Terms and senses	68.03%	79.26%	73.21%
Russian-English			
Terms only	68.36%	66.34%	67.34%
Terms and senses	73.56%	80.29%	76.78%
Russian-Italian			
Terms only	67.85%	57.48%	62.24%
Terms and senses	71.38%	72.21%	71.79%

ple approach that foregoes disambiguation: For every single occurrence of a term t, we take all sense nodes n_s with a matching part-of-speech tag, and normalize by dividing by the sum of their `meaning` arc weights. Thus, if a term has four equally relevant sense nodes in UWN, then each receives a local weight of $\frac{1}{4}$. Additionally, these senses pass on their weight to neighbouring nodes immediately connected via `hypernymy` arcs. Summing up the weights of local occurrences of a token t (either an original document term or a sense node) within a document d, one arrives at document-level occurrence scores $n(t, d)$, from which one can then compute TF-IDF feature vectors using the following formula:

$$n(t,d) \log\left(\frac{|D|}{|\{d \in D \mid n(t,d) \geq 1\}|}\right) \qquad (13)$$

where D is the set of training documents.

This approach was tested using a cross-lingual dataset derived from the Reuters RCV1 and RCV2 collections of newswire articles [31, 32]. These articles are mostly business related, and have topical class labels such as '*accounts/earnings*', '*economic performance*' or '*funding/capital*'. For several pairs of languages, we created independent datasets by randomly selecting 10 topics covered by both languages in order to arrive at $\binom{10}{2} = 45$ separate binary classification tasks, each based on 150 training documents in one language, and 150 test documents in a second language, likewise randomly selected with balanced class distributions.

Table 9 compares the standard bag-of-words TF-IDF representation for terms (using only genuine term frequencies as $n(t, d)$ in Equation 13) with the extended representation that includes mappings to sense nodes as frequencies. The scores shown were produced with linear kernel SVMs using the SVMlight implementation in its default settings, which are known to work well for text classification [23] – LIBSVM produced similar margins between the two approaches but overall slightly lower absolute scores. Since many of the Reuters topic categories are business-related, using only the original document terms, which include names of companies and people, already works surprisingly well. By considering sense nodes, both precision and recall are boosted significantly. This shows e.g. that English terms in the training set are being mapped to the same senses as the corresponding Russian terms in the test documents. The margins could be boosted even further by invoking more intelligent word sense disambiguation strategies or using more advanced sense expansion strategies. [10].

6. CONCLUSION

We have presented a novel approach to building a large-scale universal wordnet (UWN) that contains 1.5 million `meaning` relationships from over 800,000 terms in over 200 languages. UWN is available at `http://www.mpii.de/yago-naga/uwn/`. Our experiments have shown that UWN is useful in applied tasks. In addition to the existing applications of WordNet, such as question answering, text classification, semantic relatedness assessment, and so on, which are now possible for a greater range of languages, we also anticipate UWN being used for tasks that explicitly make use of multilingual connections in the network, e.g. cross-lingual information retrieval or cross-lingual text classification.

We have created a public querying and editing website for UWN that in the long run may allow us to address issues such as correcting inaccurate arcs and adding new senses to cope with language-specific subtleties (in particular lexical gaps, incongruence). Since the confidence estimates derived from the learnt models correlate quite well with the evaluated precision on the arcs, manual efforts could be channelled to focus explicitly on arcs with borderline confidence values and terms without accepted meaning arcs. An update submitted to the Web interface or an additionally imported translation dictionary for one language can subsequently lead to a sufficient amount of accumulated evidence to sway the model towards accepting mappings in entirely different languages. Hence, it is safe to expect continued growth and refinement in the future.

Finally, we envision new data-driven techniques that automatically expand the sense inventory of UWN. Snow et al. [35] have shown that this is feasible by extending WordNet using monolingual corpora. Using our universal wordnet as the underlying core, improved algorithms are conceivable.

7. REFERENCES

[1] E. Adar, M. Skinner, and D. S. Weld. Information arbitrage across multi-lingual wikipedia. In *Proc. 2nd ACM WSDM 2009*, New York, NY, USA, 2009. ACM.

[2] J. Atserias, S. Climent, X. Farreres, G. Rigau, and H. Rodríguez. Combining multiple methods for the automatic construction of multilingual WordNets. In *Proc. RANLP 1997*, 1997.

[3] J. Bos and K. Markert. Recognising textual entailment with logical inference. In *Proc. HLT/EMNLP 2005*, Morristown, NJ, USA, 2005. ACL.

[4] P. Buitelaar, T. Eigner, and T. Declerck. OntoSelect: A dynamic ontology library with support for ontology selection. In *Proc. ISWC 2004*, 2004.

[5] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines, 2001.

[6] D. Cook. Automatic translation for MLSN. In `http://dcook.org/mlsn/about/`, 2008.

[7] J. Daudé, L. Padró, and G. Rigau. Mapping wordnets using structural information. In *Proc. ACL 2000*, pages 504–511, Morristown, NJ, USA, 2000.

[8] J. Daudé, L. Padró, and G. Rigau. Making wordnet mappings robust. In *Proc. 19th Congreso de la Sociedad Española para el Procesamiento del Lenguage Natural (SEPLN)*, Universidad de Alcalá de Henares. Madrid, Spain, 2003.

[9] M. Davis and M. Dürst. Unicode normalization forms, Rev. 29. Technical report, Unicode, 2008.

[10] G. de Melo and S. Siersdorfer. Multilingual text classification using ontologies. In *Proc. ECIR 2007*, volume 4425 of *LNCS*, Rome, Italy, 2007. Springer.

[11] G. de Melo and G. Weikum. A machine learning approach to building aligned wordnets. In *Proc. International Conference on Global Interoperability for Language Resources*, 2008.

[12] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, November 2000.

[13] O. Etzioni, K. Reiter, S. Soderland, and M. Sammer. Lexical translation with application to image search on the Web. In *Proc. Machine Translation Summit XI, 2007*, 2007.

[14] C. Fellbaum, editor. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, 1998.

[15] C. Fellbaum and P. Vossen. Connecting the universal to the specific: Towards the Global Grid. In *1st International Workshop on Intercultural Collaboration, IWIC 2007*, volume 4568 of *LNCS*. Springer, 2007.

[16] D. Fišer. Using multilingual resources for building SloWNet faster. In *Proc. 4th International WordNet Conference (GWC)*, Szeged, Hungary, 2008.

[17] L. Getoor and B. Taskar. *Introduction to Statistical Relational Learning*. The MIT Press, 2007.

[18] Z. Gong, C. W. Cheang, and L. H. U. Web query expansion by WordNet. In *Proc. DEXA 2005, Copenhagen, Denmark*, volume 3588 of *LNCS*, pages 166–175. Springer, 2005.

[19] I. Gurevych, C. Müller, and T. Zesch. What to be? - Electronic career guidance based on semantic relatedness. In *Proc. ACL 2007*, Prague, Czech Republic, 2007.

[20] S. M. Harabagiu, editor. *Proc. Workshop Usage of WordNet in Natural Language Processing Systems*. ACL, Université de Montŕeal, Montŕeal, QC, Canada, 1998.

[21] H. Isahara, F. Bond, K. Uchimoto, M. Utiyama, and K. Kanzaki. Development of the Japanese WordNet. In E. L. R. A. (ELRA), editor, *Proc. LREC 2008*, Marrakech, Morocco, 2008.

[22] Y. R. Jean-Mary and M. R. Kabuka. Asmov: Results for OAEI 2008. In *Proc. 3rd International Workshop on Ontology Matching (OM-2008)*, volume 431 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.

[23] T. Joachims. Making large-scale support vector machine learning practical. In *Advances in Kernel Methods: Support Vector Machines*. MIT Press, Cambridge, MA, USA, 1999.

[24] K. Knight and S. K. Luk. Building a large-scale knowledge base for machine translation. In *Proc. AAAI 1994 (vol. 1)*, pages 773–778, Menlo Park, CA, USA, 1994.

[25] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, 2007.

[26] A. Marchetti, M. Tesconi, F. Ronzano, M. Rosella, and F. Bertagna. Toward an architecture for the Global Wordnet initiative. In *Proc. 3rd Italian Semantic Web Workshop, SWAP 2006*. CEUR-WS.org, 2006.

[27] M. Marszałek and C. Schmid. Semantic hierarchies for visual object recognition. In *IEEE Conference on Computer Vision & Pattern Recognition*, June 2007.

[28] F. J. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[29] A. Okumura and E. Hovy. Building Japanese-English dictionary based on ontology for machine translation. In *Proc. Workshop on Human Language Technology*, pages 141–146. Association for Computational Linguistics, 1994.

[30] J. C. Platt. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In A. S. *et al.*, editor, *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, Cambridge, MA, USA, 2000.

[31] Reuters. Reuters Corpus, vol. 1: English Language, 1996-08-20 to 1997-08-19, 2000.

[32] Reuters. Reuters Corpus, vol. 2: Multilingual Corpus, 1996-08-20 to 1997-08-19, 2000.

[33] N. Schlaefer, J. Ko, J. Betteridge, M. Pathak, E. Nyberg, and G. Sautter. Semantic extensions of the Ephyra QA system for TREC 2007. In *Proc. TREC 2007*. NIST, 2007.

[34] H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *Intl. Conference on New Methods in Language Processing*, Manchester, UK, 1994.

[35] R. Snow, D. Jurafsky, and A. Y. Ng. Semantic taxonomy induction from heterogenous evidence. In *Proc. ACL 2006*, pages 801–808, Morristown, NJ, USA, 2006. ACL.

[36] F. M. Suchanek, G. Kasneci, and G. Weikum. YAGO: A Core of Semantic Knowledge. In *Proc. WWW 2007*, New York, NY, USA, 2007. ACM Press.

[37] J. Tiedemann. Combining clues for word alignment. In *Proc. EACL 2003*, pages 339–346, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[38] J. Tiedemann. The OPUS corpus - parallel & free. In *Proc. LREC 2004*, 2004.

[39] D. Tufiş, D. Cristea, and S. Stamou. BalkaNet: Aims, Methods, Results and Perspectives. A General Overview. *Romanian Journal on Information Science and Technology*, 7(1–2):9–34, 4 2004.

[40] P. Vossen, editor. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Springer, 1998.