A Graph-Theoretic Algorithm for Automatic Extension of Translation Lexicons

Beate Dorow Florian Laws Lukas Michelbacher Christian Scheible Jason Utt
Institute for Natural Language Processing
Universität Stuttgart
{dorowbe,lawsfn,michells,scheibcn,uttjn}@ims.uni-stuttgart.de

Abstract

This paper presents a graph-theoretic approach to the identification of yet-unknown word translations. The proposed algorithm is based on the recursive SimRank algorithm and relies on the intuition that two words are similar if they establish similar grammatical relationships with similar other words. We also present a formulation of SimRank in matrix form and extensions for edge weights, edge labels and multiple graphs.

1 Introduction

This paper describes a cross-linguistic experiment which attempts to extend a given translation dictionary with translations of novel words.

In our experiment, we use an English and a German text corpus and represent each corpus as a graph whose nodes are words and whose edges represent grammatical relationships between words. The corpora need not be parallel.

Our intuition is that a node in the English and a node in the German graph are similar (that is, are likely to be translations of one another), if their neighboring nodes are. Figure 1 shows part of the English and the German word graph.

Many of the (first and higher order) neighbors of *food* and *Lebensmittel* translate to one another (marked by dotted lines), indicating that *food* and *Lebensmittel*, too, are likely mutual translations.

Our hypothesis yields a recursive algorithm for computing node similarities based on the similarities of the nodes they are connected to. We initialize the node similarities using an English-German dictionary whose entries correspond to known pairs of equivalent nodes (words). These node equivalences constitute the "seeds" from which novel English-German node (word) correspondences are bootstrapped.

We are not aware of any previous work using a measure of similarity between nodes in graphs for cross-lingual lexicon acquisition.

Our approach is appealing in that it is language independent, easily implemented and visualized, and readily generalized to other types of data.

Section 2 is dedicated to related research on the automatic extension of translation lexicons. In Section 3 we review SimRank (Jeh and Widom, 2002), an algorithm for computing similarities of nodes in a graph, which forms the basis of our work. We provide a formulation of SimRank in terms of simple matrix operations which allows an efficient implementation using optimized matrix packages. We further present a generalization of SimRank to edge-weighted and edge-labeled graphs and to inter-graph node comparison.

Section 4 describes the process used for building the word graphs. Section 5 presents an experiment for evaluating our approach to bilingual lexicon acquisition. Section 6 reports the results. We present our conclusions and directions for future research in Section 7.

2 Related Work on cross-lingual lexical acquisition

The work by Rapp (1999) is driven by the idea that a word and its translation to another language are likely to co-occur with similar words. Given a German and an English corpus, he computes two word-by-word co-occurrence matrices, one for each language, whose columns span a vector space representing the corresponding corpus.

In order to find the English translation of a German word, he uses a base dictionary to translate all known column labels to English. This yields a new vector representation of the German word in the English vector space. This mapped vector is then compared to all English word vectors, the most similar ones being candidate translations.

Proceedings of the EACL 2009 Workshop on GEMS: GEometical Models of Natural Language Semantics, pages 91–95,
Athens, Greece, 31 March 2009. ©2009 Association for Computational Linguistics

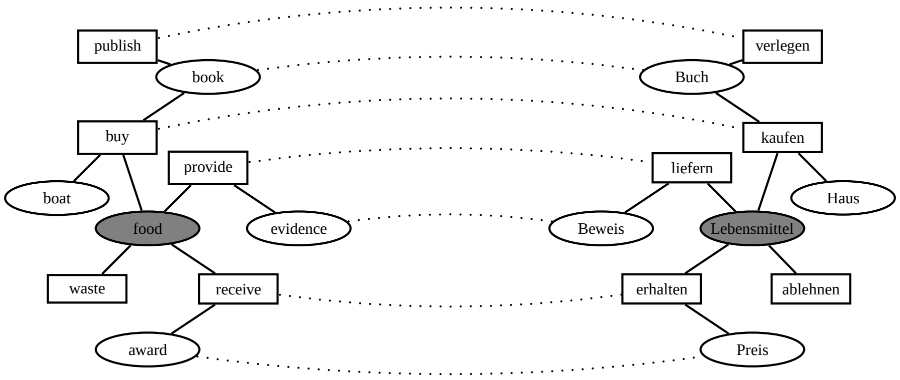

Figure 1: Likely translations based on neighboring nodes

Rapp reports an accuracy of 72% for a small number of test words with well-defined meaning.

Diab and Finch (2000) first compute word similarities within each language corpus separately by comparing their co-occurrence vectors. Their challenge then is to derive a mapping from one language to the other (i.e. a translation lexicon) which best preserves the intra-language word similarities. The mapping is initialized with a few seed "translations" (punctuation marks) which are assumed to be common to both corpora.

They test their method on two corpora written in the same language and report accuracy rates of over 90% on this pseudo-translation task. The approach is attractive in that it does not require a seed lexicon. A drawback is its high computational cost.

Koehn and Knight (2002) use a (linear) combination of clues for bootstrapping an English-German noun translation dictionary. In addition to similar assumptions as above, they consider words to be likely translations of one another if they have the same or similar spelling and/or occur with similar frequencies. Koehn and Knight reach an accuracy of 39% on a test set consisting of the 1,000 most frequent English and German nouns. The experiment excludes verbs whose semantics are more complex than those of nouns.

Otero and Campos (2005) extract English-Spanish pairs of lexico-syntactic patterns from a small parallel corpus. They then construct context vectors for all English and Spanish words by recording their frequency of occurrence in each of these patterns. English and Spanish vectors thus reside in the same vector space and are readily compared.

The approach reaches an accuracy of 89% on a test set consisting of 100 randomly chosen words from among those with a frequency of 100 or higher. The authors do not report results for low-frequency words.

3 The SimRank algorithm

An algorithm for computing similarities of nodes in graphs is the SimRank algorithm (Jeh and Widom, 2002). It was originally proposed for directed unweighted graphs of web pages (nodes) and hyperlinks (links).

The idea of SimRank is to recursively compute node similarity scores based on the scores of neighboring nodes. The similarity S_{ij} of two different nodes i and j in a graph is defined as the normalized sum of the pairwise similarities of their neighbors:

$$S_{ij} = \frac{c}{|N(i)|\ |N(j)|} \sum_{k \in N(i), l \in N(j)} S_{kl}. \qquad (1)$$

$N(i)$ and $N(j)$ are the set of i's and j's neighbors respectively, and c is a multiplicative factor smaller than but close to 1 which demotes the contribution of higher order neighbors. S_{ij} is set to 1 if i and j are identical, which provides a basis for the recursion.

3.1 Matrix formulation of SimRank

We derive a formulation of the SimRank similarity updates which merely consists of matrix multiplications as follows. In terms of the graph's (binary) adjacency matrix A, the SimRank recursion reads:

$$S_{ij} = \frac{c}{|N(i)|\ |N(j)|} \sum_{k \in N(i), l \in N(j)} A_{ik}\, A_{jl}\, S_{kl} \qquad (2)$$

noting that $A_{ik}A_{jl} = 1$, iff k is a neighbor of i and l is a neighbor of j at the same time. This is equivalent to

$$\begin{aligned} S_{ij} &= c \sum_{k,l} \frac{A_{ik}}{|N(i)|} \frac{A_{jl}}{|N(j)|} S_{kl} \qquad (3) \\ &= c \sum_{k,l} \frac{A_{ik}}{\sum_\nu A_{i\nu}} \frac{A_{jl}}{\sum_\nu A_{j\nu}} S_{kl}. \end{aligned}$$

The S_{ij} can be assembled in a square node similarity matrix S, and it is easy to see that the individual similarity updates can be summarized as:

$$S_k = c\, \tilde{A}\, S_{k-1} \tilde{A}^T \qquad (4)$$

where $\tilde{A}$ is the row-normalized adjacency matrix and k denotes the current level of recursion. $\tilde{A}$ is obtained by dividing each entry of A by the sum of the entries in its row. The SimRank iteration is initialized with $S = I$, and the diagonal of S, which contains the node self-similarities, is reset to ones after each iteration.

This representation of SimRank in closed matrix form allows the use of optimized off-the-shelf sparse matrix packages for the implementation of the algorithm. This rendered the pruning strategies proposed in the original paper unnecessary. We also note that the Bipartite SimRank algorithm introduced in (Jeh and Widom, 2002) is just a special case of Equation 4.

3.2 Extension with weights and link types

The SimRank algorithm assumes an unweighted graph, i.e. a binary adjacency matrix A. Equation 4 can equally be used to compute similarities in a *weighted* graph by letting $\tilde{A}$ be the graph's row-normalized *weighted* adjacency matrix. The entries of $\tilde{A}$ then represent transition probabilities between nodes rather than hard (binary) adjacency. The proof of the existence and uniqueness of a solution to this more general recursion proceeds in analogy to the proof given in the original paper.

Furthermore, we allow the links in the graph to be of different types and define the following generalized SimRank recursion, where $\mathcal{T}$ is the set of link types and $N_t(i)$ denotes the set of nodes connected to node i via a link of type t.

$$S_{ij} = \frac{c}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \frac{1}{|N_t(i)|\ |N_t(j)|} \sum_{k \in N_t(i), l \in N_t(j)} S_{kl}. \qquad (5)$$

In matrix formulation:

$$S_k = \frac{c}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \tilde{A}_t\, S_{k-1} \tilde{A}_t^{\,T} \qquad (6)$$

where A_t is the adjacency matrix associated with link type t and, again, may be weighted.

3.3 SimRank across graphs

SimRank was originally designed for the comparison of nodes within a single graph. However, SimRank is readily and accordingly applied to the comparison of nodes of two different graphs. The original SimRank algorithm starts off with the nodes' self-similarities which propagate to other non-identical pairs of nodes. In the case of two different graphs $\mathcal{A}$ and $\mathcal{B}$, we can instead initialize the algorithm with a set of initially known node-node correspondences.

The original SimRank equation (2) then becomes

$$S_{ij} = \frac{c}{|N(i)|\ |N(j)|} \sum_{k,l} A_{ik}\, B_{jl}\, S_{kl}, \qquad (7)$$

which is equivalent to

$$S_k = c\, \tilde{A}\, S_{k-1}\, \tilde{B}^T, \qquad (8)$$

or, if links are typed,

$$S_k = \frac{c}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \tilde{A}_t\, S_{k-1}\, \tilde{B}_t^{\,T}. \qquad (9)$$

The similarity matrix S is now a rectangular matrix containing the similarities between nodes in $\mathcal{A}$ and nodes in $\mathcal{B}$. Those entries of S which correspond to known node-node correspondences are reset to 1 after each iteration.

4 The graph model

The grammatical relationships were extracted from the British National Corpus (BNC) (100 million words), and the Huge German Corpus (HGC) (180 million words of newspaper text). We compiled a list of English verb-object (V-O) pairs based on the verb-argument information extracted by (Schulte im Walde, 1998) from the BNC. The German V-O pairs were extracted from a syntactic analysis of the HGC carried out using the BitPar parser (Schmid, 2004).

We used only V-O pairs because they constitute far more sense-discriminative contexts than, for example, verb-subject pairs, but we plan to examine these and other grammatical relationships in future work.

We reduced English compound nouns to their heads and lemmatized all data. In English phrasal

English						German					
Low		Mid		High		Low		Mid		High	
N	V	N	V	N	V	N	V	N	V	N	V
0.313	0.228	0.253	0.288	0.253	0.255	0.232	0.247	0.205	0.237	0.211	0.205

Table 1: The 12 categories of test words, with mean relative ranks of test words

verbs, we attach the particles to the verbs to distinguish them from the original verb (e.g *put off* vs. *put*). Both the English and German V-O pairs were filtered using stop lists consisting of modal and auxiliary verbs as well as pronouns. To reduce noise, we decided to keep only those relationships which occurred at least three times in the respective corpus.

The English and German data alike are then represented as a bipartite graph whose nodes divide into two sets, verbs and nouns, and whose edges are the V-O relationships which connect verbs to nouns (cf. Figure 1). The edges of the graph are weighted by frequency of occurrence.

We “prune” both the English and German graph by recursively removing all leaf nodes (nodes with a single neighbor). As these correspond to words which appear only in a single relationship, there is only limited evidence of their meaning.

After pruning, there are 4,926 nodes (3,365 nouns, 1,561 verbs) and 43,762 links in the English, and 3,074 nodes (2,207 nouns, 867 verbs) and 15,386 links in the German word graph.

5 Evaluation experiment

The aim of our evaluation experiment is to test the extended SimRank algorithm for its ability to identify novel word translations given the English and German word graph of the previous section and an English-German seed lexicon. We use the dict.cc English-German dictionary [1].

Our evaluation strategy is as follows. We select a set of test words at random from among the words listed in the dictionary, and remove their entries from the dictionary. We run six iterations of SimRank using the remaining dictionary entries as the seed translations (the known node equivalences), and record the similarities of each test word to its known translations. As in the original SimRank paper, c is set to 0.8.

We include both English and German test words and let them vary in frequency: high- ($>$ 100), mid- ($>$ 20 and $\leq$ 100), and low- ($\leq$ 20) frequent as well as word class (noun, verb). Thus, we obtain 12 categories of test words (summarized in Table 1), each of which is filled with 50 randomly selected words, giving a total of 600 test words.

SimRank returns a matrix of English-German node-node similarities. Given a test word, we extract its row from the similarity matrix and sort the corresponding words by their similarities to the test word. We then scan this sorted list of words and their similarities for the test word’s reference translations (those listed in the original dictionary) and record their positions (i.e. ranks) in this list. We then replace absolute ranks with relative ranks by dividing by the total number of candidate translations.

6 Results

Table 1 lists the mean relative rank of the reference translations for each of the test categories. The values of around 0.2-0.3 clearly indicate that our approach ranks the reference translations much higher than a random process would.

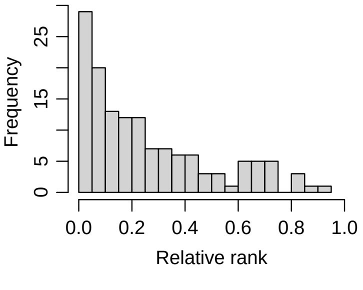

Figure 2: Distribution of the relative ranks of the reference translations in the English-High-N test set.

Exemplary of all test sets, Figure 2 shows the distribution of the relative ranks of the reference translations for the test words in English-High-N. The bulk of the distribution lies below 0.3, i.e. in the top 30% of the candidate list.

In order to give the reader an idea of the results, we present some examples of test words and their

[1] http://www.dict.cc/ (May 5th 2008)

Test word	Top 10 predicted translations	Ranks

sanction	Ausgangssperre Wirtschaftssanktion Ausnahmezustand Embargo Moratorium *Sanktion* Todesurteil Geldstrafe Bußgeld Anmeldung	Sanktion(6)
		Maßnahme(1407)

delay	anfechten revidieren zurückstellen füllen verkünden quittieren vertagen verschieben aufheben respektieren	verzögern(78)
		aufhalten(712)

Kosten	hallmark trouser blouse makup uniform armour robe testimony witness jumper	cost(285)

öffnen	unlock lock usher step peer shut guard hurry slam close	open(12)
		undo(481)

Table 2: Some examples of test words, their predicted translations, and the ranks of their true translations.

predicted translations in Table 2.

Most of the 10 top-ranked candidate translations of *sanction* are hyponyms of the correct translations. This is mainly due to insufficient noun compound analysis. Both the English and German nouns in our graph model are single words. Whereas the English nouns consist only of head nouns, the German nouns include many compounds (as they are written without spaces), and thus tend to be more specific.

Some of the top candidate translations of *delay* are correct (*verschieben*) or at least acceptable (*vertagen*), but do not count as such as they are missing in the gold standard dictionary.

The mistranslation of the German noun *Kosten* is due to semantic ambiguity. *Kosten* co-occurs often with the verb *tragen* as in *to bear costs*. The verb *tragen* however is ambiguous and may as well be translated as *to wear* which is strongly associated with clothes.

We find several antonyms of *öffnen* among its top predicted translations. Verb-object relationships alone do not suffice to distinguish synonyms from antonyms. Similarly, it is extremely difficult to differentiate between the members of closed categories (e.g. the days of the week, months of the year, mass and time units) using only syntactic relationships.

7 Conclusions and Future Research

The matrix formulation of the SimRank algorithm given in this paper allows an implementation using efficient off-the-shelf software libraries for matrix computation.

We presented an extension of the SimRank algorithm to edge-weighted and edge-labeled graphs. We further generalized the SimRank equations to permit the comparison of nodes from two different graphs, and proposed an application to bilingual lexicon induction.

Our system is not yet accurate enough to be used for actual compilation of translation dictionaries. We further need to address the problem of data sparsity. In particular, we need to remove the bias towards low-degree words whose similarities to other words are unduly high.

In order to solve the problem of ambiguity, we intend to apply SimRank to the incidence representation of the word graphs, which is constructed by putting a node on each link. The proposed algorithm will then naturally return similarities between the more sense-discriminative links (syntactic relationships) in addition to similarities between the often ambiguous nodes (isolated words).

References

M. Diab and S. Finch. 2000. A statistical word-level translation model for comparable corpora. In *In Proceedings of the Conference on Content-Based Multimedia Information Access (RIAO)*.

G. Jeh and J. Widom. 2002. Simrank: A measure of structural-context similarity. In *KDD '02: Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 538–543.

P. Koehn and K. Knight. 2002. Learning a translation lexicon from monolingual corpora. In *Proceedings of the ACL-02 Workshop on Unsupervised Lexical Acquisition*, pages 9–16.

P. Gamallo Otero and J. Ramon Pichel Campos. 2005. An approach to acquire word translations from non-parallel texts. In *EPIA*, pages 600–610.

R. Rapp. 1999. Automatic identification of word translations from unrelated English and German corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 519–526.

Helmut Schmid. 2004. Efficient parsing of highly ambiguous context-free grammars with bit vectors. In *COLING '04: Proceedings of the 20th International Conference on Computational Linguistics*, page 162.

Sabine Schulte im Walde. 1998. Automatic Semantic Classification of Verbs According to Their Alternation Behaviour. Master's thesis, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.