Cross-Dictionary Linking at Sense Level with a Double-Layer Classifier

Roser Saurí
Dictionaries Technology Group, Oxford University Press, UK
roser.sauri@oup.com

Louis Mahon
Dictionaries Technology Group, Oxford University Press, UK
Oxford University, UK
louis.mahon@linacre.ox.ac.uk

Irene Russo
Dictionaries Technology Group, Oxford University Press, UK
ILC *A. Zampolli* - CNR, Pisa, Italy
irene.russo@ilc.cnr.it

Mironas Bitinis
Dictionaries Technology Group, Oxford University Press, UK
mkbitinis@gmail.com

Abstract

We present a system for linking dictionaries at the sense level, which is part of a wider programme aiming to extend current lexical resources and to create new ones by automatic means. One of the main challenges of the sense linking task is the existence of non one-to-one mappings among senses. Our system handles this issue by addressing the task as a binary classification problem using standard Machine Learning methods, where each sense pair is classified independently from the others. In addition, it implements a second, statistically-based classification layer to also model the dependence existing among sense pairs, namely, the fact that a sense in one dictionary that is already linked to a sense in the other dictionary has a lower probability of being linked to a further sense. The resulting double-layer classifier achieves global Precision and Recall scores of 0.91 and 0.80, respectively.

2012 ACM Subject Classification Computing methodologies → Lexical semantics; Computing methodologies → Language resources; Computing methodologies → Supervised learning by classification

Keywords and phrases Word sense linking, word sense mapping, lexical translation, lexical resources, language data construction, multilingual data, data integration across languages

Digital Object Identifier 10.4230/OASIcs.LDK.2019.20

Acknowledgements We are very grateful to Charlotte Buxton and Rebecca Juganaru, the expert lexicographers who have contributed all the dictionary knowledge we were lacking and have helped with manual annotations. In addition, we want to express our thanks to Richard Shapiro, Will Hunter and Sophie Wood for their great support in different aspects of the project. All errors and mistakes are responsibility of the authors.

1 Introduction

Dictionary usage has changed tremendously in the past decades, both in terms of quality (e.g., type of searches, preferred support: paper or digital, etc.) and quantity (number of dictionary users, average number of searches by user, etc.). That dictionaries as a product are in decline is a well-known fact, but this trend is not appreciated in the case of bilingual dictionaries. In spite of the availability of free translation tools of remarkable quality, often integrated in web browsers, the generalization of internet access paired with the growth of online content in multiple languages seems to guarantee their continuance.

2nd Conference on Language, Data and Knowledge (LDK 2019).
Editors: Maria Eskevich, Gerard de Melo, Christian Fäth, John P. McCrae, Paul Buitelaar, Christian Chiarcos, Bettina Klimek, and Milan Dojchinovski; Article No. 20; pp. 20:1–20:16
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

An obvious and very widespread use case for bilingual dictionaries is supporting second language learning. Although learners can nowadays resort to online content for a quick translation, manually edited dictionaries remain the go-to sources for good quality information, especially with respect to less frequent uses, and wider descriptions on how words are employed. This is because dictionaries filter out noisy content, distill the key aspects of linguistic expressions, and provide a broad view on words, e.g., labels for register or domain.

Bilingual dictionaries also play a key role in several language technology areas. For instance, they are a component of search engines for cross-lingual information retrieval, or in metadata tagging tools for multilingual image search systems. Moreover, they are complementary to machine translation systems, which despite their significant improvement with the advent of neural networks technology in the past years, still fall short of returning adequate or informative enough answers when it comes to translating words or lexical constructions provided out of context.

The manual compilation of dictionaries is nevertheless a costly and time-consuming activity, which has led to efforts towards developing methods for (semi-)automating the process. An example of this is the shared task *Translation Inference Across Dictionaries* (TIAD), initiated in 2017 with the aim of exploring methods and techniques to auto-generate bilingual and multilingual dictionaries based on existing ones.[1] The current paper presents research in a similar direction. In particular, it introduces a piece of work embedded within a wider programme with a two-fold goal:

1. Automatically creating new bilingual dictionaries, a task that touches upon the area known as *lexical translation*, concerning systems able to return translations of words or phrases, e.g., [15].
2. Enriching existing bilingual dictionary information with additional data available from other lexical resources (e.g., sense definitions, grammatical notes, domain information, etc.). This second task has to do with the area referred to as *word sense linking* (aka *sense alignment*, *sense mapping* or *sense matching*) [10].

To these ends, we developed a system for linking entry senses from a monolingual dictionary in language L and a bilingual dictionary between languages L and L' whenever they correspond to the same meaning. In particular, we considered sense links between a monolingual English dictionary and the English side of an English-L' bilingual dictionary.

Linking senses from a bilingual dictionary to a monolingual one is the first step towards goal 2 above of enriching the content of bilingual sources, given that monolingual dictionaries tend to offer information of a different nature from that available in bilingual dictionaries. Furthermore, this same sense linking component can feed into a broader system for developing new bilingual dictionaries. By taking the monolingual dictionary as the pivot to which several bilinguals are linked at the sense level, we expect to be able to automate the creation of bilingual dictionaries involving language pairs not covered by the original bilinguals, therefore addressing goal 1 above. The process is illustrated in Figure 1.

Given an initial phase (Step 1) where the senses in the English side of the bilingual dictionaries are linked to the corresponding senses in the English monolingual dictionary, it should be possible to then move to a second phase (Step 2) where the English senses act as the bridge between the non-English parts of the two bilinguals, thus generating a bilingual dictionary for a new language pair. This paper focuses on the work carried out for Step 1.

[1] See: https://tiad2017.wordpress.com/ and http://tiad2019.unizar.es

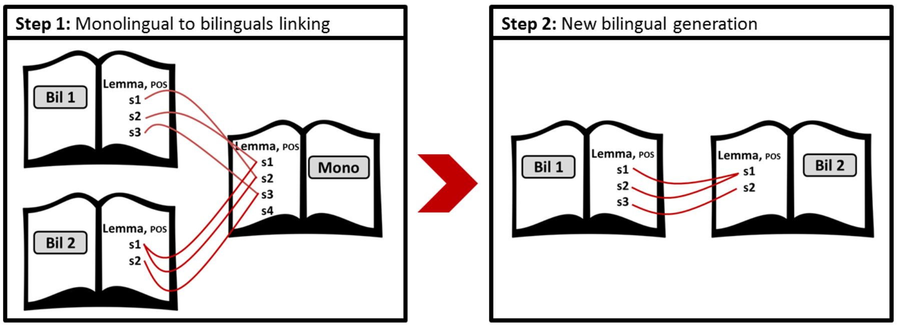

Figure 1 Automated bilingual dictionary generation process.

One of the main challenges of the sense linking task is the fact that it is not restricted to one-to-one mappings. Dictionaries differ in terms of *sense granularity* (that is, one sense in a dictionary corresponds to two or more in another), and in terms of *coverage* (one sense in a dictionary does not correlate to any in the other). Throughout the paper we will refer to this type of misalignment as *non one-to-one mappings.* A further challenge, in this case specific to our project, has to do with the different nature of information in bilinguals as opposed to monolinguals. While the latter tend to contain more extensive textual elements, bilinguals do not have definitions but describe senses by means of translations or short glosses. We will show that the system we put forward here offers a solution to these two issues.

The paper is structured as follows. Section 2 discusses related work. Then, sections 3 and 4 describe the solution proposed to the task at hand. In the first of these sections we give a global overview on the methodology we followed, while the second one goes into the design details of the system we developed. Results are presented in Section 5, and Section 6 closes with final remarks and suggests directions for future work.

2 Related Work

The work presented here belongs to the area of *sense linking* and also, although less directly, to that of *lexical translation.* Less directly in the latter case because, as just argued, the development of a full lexical translation system has yet to be completed. In spite of that, we considered it worth reviewing previous work also on that second area.

Sense linking. The past years have witnessed notable activity in this field, motivated by the interest in developing large Linked Lexical Knowledge Bases (LLKBs) by means of integrating multiple resources into a single one (e.g., BabelNet [17], UBY [9]) in order to achieve maximum lexical coverage and information richness, and thus to be able to better support different NLP tasks. Most of this previous activity involves direct sense linking of Lexical Knowledge Bases (LKBs), as opposed to more traditional dictionary content, even if shaped as Machine Readable Dictionaries (MRDs), e.g., Niemann and Gurevych [18] among many others. The difference between dictionaries (or MRDs) and LKBs is that the latter organize their content in a graph-based structure, depicting the lexical relations that hold among words (e.g., hyper- and hyponymy, entailment, synonymy, etc.). Thus, much of the research on LKB sense linking benefits from lexical information structural organization.

Nevertheless, there is also some work around sense linking which disregards information organization structure and is based solely on similarity between textual elements such as definitions. This approach appears more suited for sense linking dictionary content, although as will be seen next, in some cases it has been applied to LKBs only. A first strategy here relies on *word overlap*, that is, on the number of words shared by the textual elements in each dictionary, e.g., the early work by Lesk [13] and Byrd [2]. More recently also, Ponzetto and Navigli [19] used word overlap for a conditional probability-based approach for aligning Wordnet and Wikipedia. There are some significant shortcomings of this strategy: it strongly depends on the presence of common words, and in some cases the number of shared words is the same for different senses of the same entry, making the decision hard.

A second, more elaborate strategy consists in representing dictionary textual elements as *vectors in a multi-dimensional vector space* and then computing the distance between them as a proxy for their similarity. The closer the vectors, the more similar the texts they represent. Ruiz-Casado and colleagues [21], for example, followed this strategy for sense aligning Wikipedia articles to their corresponding WordNet synsets [6]. Nevertheless, two major drawbacks of this strategy are, first, the need to set a threshold for determining equivalent senses; and second, the fact that only one-to-one mappings can be accounted for, while it is often the case that a sense in one dictionary corresponds to several in the other.

These issues are not shared by other research resorting to well-known *graph-based methods* for modelling textual information. For example, Ide and Veronis [11] built a complex network of senses and the words present in their definitions, and applied a spreading activation strategy for identifying sense pairs between the *Oxford Advanced Learner's Dictionary* (OALD) and the *Collins English Dictionary* (CED). Although the authors reported good results (90% accuracy), the experiments were unfortunately quite partial as they were applied to only 59 senses selected from OALD. What is more relevant for us here is the fact that the proposed system, seemingly successful when applied to two monolingual dictionaries, does not appear suitable for linking a monolingual and a bilingual dictionary, given that the latter does not contain sense definitions but only translations and indicators.

To our knowledge, there is no work applying a *Machine Learning (ML)* based approach yet to the task of sense linking dictionary content This is the strategy adopted in this project because it can handle non one-to-one mappings and does not require setting any threshold.

Lexical translation. Lexical translation involves systems capable of providing translations for words or lexical expressions. It is closely related to the automatic creation of bilingual dictionary content, especially concerning languages for which there are no translation lexicons of any sort. Work in this area tends to rely on the combination of several bilingual dictionaries to generate a new one involving a language pair not covered in the initial bilingual lexicons. A basic strategy for that is known as *triangulation*. It generates new translation pairs from a source language L_{source} to a target language L_{target} by simultaneously translating from L_{source} to 2 intermediate languages, L_{inter_1} and L_{inter_2}, and from each of these to the target language L_{target}. The final translation is obtained from what is shared in both translation paths. See for example [8, 14].

A second strategy is based on *translation cycles* across languages (as in, e.g., [24, 1]). A cycle is a translation chain across different languages which starts and ends with the same term. For instance, $t_{L_1} > t_{L_2} > t_{L_3} > t_{L_1}$, where t_L is the term used for a word in language L, and $t_L > t_{L'}$ expresses that term t_L translates to $t_{L'}$. Not all translation chains correspond to translation cycles due to the semantic shift that may take place between translations (e.g., a word in one language can have a wider or narrower meaning than its translation in another). Thus, this approach considers as valid only the translation pairs within a translation cycle. Translation cycles tend to give a good precision score because the cycle guarantees translation validity, but low coverage due to its restrictiveness.

Finally, a third approach is based on the notion of *transitivity chains*. That is, the possibility of translating term t_A to term t_C if it is the case that $t_A > t_B$ and at the same time $t_B > t_C$. There are different takes on that, e.g., using probabilistic inference algorithms [15], supporting the decision with parallel corpora [20], or training a machine learning classifier [5].

All these approaches, however, rely exclusively on bilingual dictionaries. This means that a potential lower degree of lexical coverage in any bilingual dictionary used as intermediate step will cause the triangulation or chain to fail. Similarly, differences in sense granularity between two bilinguals may invalidate the linking with a third one. These issues can be avoided if using a more complete, finer-grained monolingual dictionary as a pivot to which to link all bilingual dictionaries. The monolingual dictionary will act as the bridge across the different languages and therefore will ensure consistency on sense equivalence [22, 4, 12, 23, 26]. Our work aligns with this other line of research.

3 Methodology

3.1 General Overview

Since we had a large amount of manually annotated data already available (see Section 3.2), we opted for a ML-based approach. Specifically, we approached the task building a binary classifier capable of judging any sense pair as a *link* (i.e., both senses correspond to the same meaning) or a *non-link* (each sense denotes something different). A sense pair is a pair (s_{mono}, s_{bil}), where s_{mono} is a sense from an entry in the monolingual dictionary and s_{bil} a sense from the same entry in the bilingual dictionary.

The requirement of both senses to belong to the same entry means that they have the same lemma and part of speech (POS) class (e.g., *water* NOUN is different from *water* VERB). We will refer this unit of information as *lexeme*. Note that dictionary homographs (e.g., lie_1 VERB "*Be in or assume a horizontal position*" vs. lie_2 VERB "*Tell a lie*") will be considered here as belonging to the same lexeme unit, thus deviating from the standard notion.

Given that the classifier considers each sense pair independently, differences of granularity do not pose a challenge anymore. Any sense in one dictionary can be linked to another sense in the other even if it has previously been linked to a further sense. This strategy, however, is not sensitive to the fact that senses already linked to a sense in the other dictionary have a lower probability of being linked to a second sense. Thus, in order to also benefit from this observation, we complemented the ML classifier with a meta-algorithm which adjusts the judgment on each sense pair based on the potential existence of other links for the same senses in the pair, as will be explained in detail in Section 4.

3.2 Dictionary Sources and Manual Annotation

We took the *Oxford Dictionary of English* (ODE)[2] as the monolingual dictionary, and linked it to the English side of several bilingual dictionaries, also compiled by Oxford University Press, involving English and a second language: English-German (EN-DE), English-Spanish (EN-ES), English-French (EN-FR), English-Italian (EN-IT), English-Russian (EN-RU), and English-Chinese (EN-ZH).[3] To our benefit, the bilingual dictionaries had already been manually linked to ODE at the sense level. The task had been performed by

[2] `https://en.oxforddictionaries.com/` (August 2017 release).

[3] `https://premium.oxforddictionaries.com/`

Table 1 Dictionary fields extracted to build the vector features, by alphabetical order, indicating the type of dictionary they belong to.

Field	Dict. Type	Description
Collocate	Bilingual	Type of words that can be collocated with the word at point (e.g., *food* is a subject collocate for *eat*). Collocates for verbs specify whether they are usually the object or the subject.
Definition	Monolingual	Description of the word meaning.
Domain	Both	Semantic area of a word (e.g., *Medicine*)
Gram. Feature	Both	Grammatical traits of the word. For example, type of complementation pattern for verbs (intransitive, transitive, etc.)
Indicator	Bilingual	Meaning description, generally a one-word or short phrase expression (e.g., *sickly sweet, sweet-tasting*, etc.)
Region	Both	Providing the geographical location of a word (e.g., *British*)
Register	Both	Classifying the tone of a word (e.g., *formal*)
Sense order	Both	Ranking of the sense within its lexeme.

expert lexicographers at Oxford University Press, who examined all senses in the bilingual dictionaries except for those: (a) tagged with the POS classes of *abbreviation* or *symbol*, and (b) presenting no information other than the translation term, i.e., lacking other possible data such as domain, register, region, collocates, example sentences, etc. These annotations were used for training the model and as gold standard to assess results (see Section 3.5).

3.3 Classifier Development Datasets

Instances creation. The dataset of instances for developing our classifier was created as follows: for each lexeme present in both ODE and the bilingual dictionary, we generated all possible sense pairs resulting from coupling each sense s_{mono} from ODE with each sense s_{bil} in the bilingual, i.e., the Cartesian product $S_{mono} \times S_{bil}$, where S_{mono} and S_{bil} are respectively the sets of monolingual and bilingual senses for that lexeme. The resulting set of sense pairs included both sense links and non-links.

Next, for each sense pair in $S_{mono} \times S_{bil}$, the dictionary fields in Table 1 were extracted together with the label *link* or *non-link* that had been manually tagged. Sense pairs for which the bilingual sense had only a translation and no other information, were excluded. The translation field was not useful for our purposes. The extracted pieces of dictionary information were used to build feature vectors, as will be explained in Section 4.1.

Splitting the dataset by POS class. Some POS classes tend to have a higher degree of polysemy than others. Verbs, for instance, are significantly more polysemous than nouns, and even more so than adverbs, as can be seen in Table 2.

Based on this observation, we experimented with separately trained models for different POS classes. We split the training set into 5 subsets, for (a) adjectives, (b) adverbs and prepositions, (c) nouns, (d) verbs, and (d) all the remainder classes (pronouns, determiners, conjunctions, interjections, etc.). The resulting sizes and their class frequencies (links, non-links) are presented in Table 3.

Table 2 Polysemic behaviour by POS class in the *Oxford Dictionary of English*: % of monosemous entries (i.e., single-sense entries), % of entries with 5 or more senses, % of entries with 10 or more senses, and maximum number of senses found in an entry for that POS class.

	% monosemous entries	% entries 5 or more senses	% entries 10 or more senses	max. no. of senses
Adjs	74.4%	2.2%	0.4%	40
Advs & Preps	83.4%	1.7%	0.4%	26
Nouns	76.8%	3.1%	0.6%	53
Verbs	51.2%	11.5%	2.6%	49
Other	72.3%	5.4%	0.7%	22

Table 3 Dataset characteristics: Number of instances, percentage of instances over the dataset, percentage of instances corresponding to links, percentage of instances corresponding to non links.

	No. instances	% instances	% links	% no links
Adjs	228,170	13.7%	37.5%	62.5%
Advs & Preps	42,369	2.5%	35.7%	64.3%
Nouns	824,503	49.6%	31.5%	68.5%
Verbs	556,969	33.5%	15.7%	84.3%
Other	11,256	0.7%	50.6%	49.4%
All POS	1,663,267	100%	27.2%	72.8%

3.4 Building the Classifier

ML classifier. The system features were engineered following recommendations from expert lexicographers from Oxford University Press, who were very acquainted with the content in the different dictionaries. We ran several rounds of experiments and assessed results using standard measures of feature importance and feature ablation techniques. Section 4.1 describes the key features in more detail, while the appendix provides the complete list. We experimented with different ML algorithms (Naïve Bayes, Support Vector Machines, Decision Trees), and based on results opted for the ensemble method Adaboost applied on DTrees.[4]

Meta-classifier. Judging each possible sense pair independently from the others allows to handle the challenges posed by non one-to-one mappings (i.e., differences of granularity and coverage). Nevertheless, sense links are to some extent dependent on the existence of other sense links in the same lexeme. That is, a sense in one dictionary already linked to a sense in the other dictionary has a lower probability of being linked to an additional sense. This observation prompted the development of a meta-classifier sensitive to the number of senses already linked in the same lexeme. We compared results from applying or not applying this algorithm on top of the ML-based classifier. Thus, we investigated two experimental settings:

- **Single-layer classifier:** Using an ML classifier only
- **Double-layer classifier:** Using an ML classifier in combination with the meta-classifier

[4] Specifically, we used python `sk-learn` implementation of [7], with parameters tree maximum depth `max_depth=1`, maximum number of estimators `n_estimators=100`, and `learning_rate=1`.

Baseline classifier. Finally, in order to evaluate the performance of each model, we compared the results against those of a baseline classifier. For each lexeme, the baseline classifier simply links the first monolingual sense to the first bilingual sense, the second to the second, and so on. Formally:

$$B((s_{mono_i}, s_{bil_j})) = 1 \iff i = j \qquad (1)$$

3.5 Evaluation

In order to avoid overfitting the model, we applied 10-fold cross validation on the manually annotated data, which thus was used as gold standard against which to assess results. Performance was scored by means of Precision, Recall and its associated F1 measure on sense pairs classified by the model as links. In addition, we used Cohen's Kappa as a way to disregard the effect of correct classifications occurring by chance.

4 Experiment Settings

This section presents the experimental settings in more detail. Specifically, Subsection 4.1 describes the features used by the ML classifier, whereas Subsection 4.2 describes the meta-classifier algorithm applied in conjunction with the ML classifier to take into account possible dependencies among sense pairs.

4.1 ML Classifier Features

In total we considered 120 features, 42 of which were selected for the final classifier. The complete list of the selected features is given in the appendix. Here we explain the rationale applied to create them. We developed two types of features: (a) based on the dictionary fields (presented in Table 1), and (b) based on the entry sense structure, i.e., the ordering of senses within each entry.

4.1.1 Features Based on Dictionary Fields

Domain, register and region. In the dictionaries we used, these three fields can be found qualifying different pieces of information, such as the definition in the monolingual dictionary, the translation in the bilingual, or some example sentences. We extracted domain, register and region elements while differentiating the piece of data they were associated to, and built independent features for each of these. There were 2 types of features based on these fields:

- Boolean features indicating whether the monolingual or bilingual sense has *domain* (or *register*, or *region*) information;
- Similarity scores (ranging [0,1]), comparing the *domain* (or *register*, or *region*) tags from the monolingual and bilingual dictionaries. Similarity was computed in one of two ways: either by the Wu-Palmer metric on WordNet [25], or by measuring how often the two tags cooccurred on the same sense in the same dictionary (note, this value is 1 if and only if both tags are the same).

A single "cross comparison" feature was also included comparing the tags from all possible locations in one dictionary (definition, example sentences, etc.) with the tags from all possible locations in the other.

Indicators and definitions. For each sense pair, the monolingual *definition* and the bilingual *indicator* were compared using two features:

- A Boolean feature indicating if they had a word in common;
- A semantic similarity score (ranging [0,1]) calculated as the cosine similarity between vectors generated with `word2vec` on the GoogleNews corpus, thus leveraging recent advances in word embedding technologies [16] to compute more accurate semantic comparisons.

Grammatical features. We built a Boolean feature for verbs only, encoding if both dictionary senses shared the same complement pattern (i.e., transitive, intransitive, etc.). Similarly, nouns had two bespoke features, signaling if the monolingual and bilingual senses shared the same countability (mass vs. count) and type (proper vs. common).

All textual fields. As a final semantic comparison, we concatenated all text fields from the bilingual sense on the one hand, and all text fields from the monolingual sense on the other, and compared the two resulting text segments using word vectors as described above.

Naive Bayes. One of the major challenges that emerged in this project was the sparsity of each feature. Because there are many possible types of dictionary information for each sense (*domain, register*, etc.) with only one or two actually being realized, the majority of features were null most of the time. The classifier, however, expected to read the same number of features for all instances, so by default it converted null values to 0, negatively impacting on its performance. Consequently, we found that the more common a feature was the more helpful it was observed to be for our performance, and so a natural course of action was to explicitly design a feature to be non-null. With this in mind, we computed a simple probability estimate using a Naive Bayes classifier on all the non-null features for a given instance, where the assumption of independence let us ignore the null values. We discretized each feature into 10 bins, and equated the conditional probabilities with the empirical probability of a link:

$$p(y=1|x_i=b)=\frac{N_{i,b,1}}{N_{i,b,0}+N_{i,b,1}} \tag{2}$$

where $N_{i,b,c}$ is the number of data points with *ith* feature equal to b, receiving classification c. The product of all such features was then added as an additional feature, $\prod_{i\in F} p_i$, where F is the set of all non-null features. At the cost of an independence assumption, this feature filtered out the noise introduced by the null values. As this Naive Bayes estimate assumes the features are class-conditionally independent, and as this does not fully hold in practice, the product in (2) is often the product of many small values and so it tends to 0. To counteract this, we worked with the geometric mean of all non-null features, instead of the product. Thus, the value for this feature was given by:

$$\sqrt[|F|]{\prod_{i\in F} p_i} \tag{3}$$

4.1.2 Features Based on Entry Sense Structure

Sense frequency. Some senses for a given entry are more frequent than others, and this partially informs how senses are ranked in an entry. That is, more common senses tend to be placed first. Based on that, we inferred an estimate of the frequency of each sense according to its position in the entry, and used the result to form a feature. We assumed that the frequency of use of a sense is a monotonically decreasing function of its position in the lexeme, and after experimenting with some obvious choices for such a function, we found $f(n) = 1/n + 1/n^2$ to give reasonable frequency estimates when evaluated qualitatively. This function was then normalized for each lexeme (the number of senses in a lexeme is variable so they must be normalized separately). The resulting feature was the absolute difference of the normalized frequency estimates for each of the two senses.

Main sense. The first listed sense in a lexeme can in general be assumed to be the most commonly used and most general. Therefore it was felt that the first sense of each dictionary could more likely (a) be linked to the first sense in the other dictionary, (b) contain multiple links than later senses in the lexeme. To supply this information to the classifier, we included two Boolean features, which indicated whether the bilingual sense and the monolingual sense were the main senses in their respective lexeme.

Single sense. We hypothesized that senses which are the only sense in their lexeme will more likely be linked to at least one sense in the other dictionary. Thus, we included two Boolean features, indicating whether the bilingual sense and the monolingual sense were single senses in their respective lexemes.

4.2 Meta-algorithm for dependent classifications

The ML classifier considers whether a sense pair within a lexeme corresponds to a link individually, without taking into consideration the existence of other links for the same senses in that pair. A complementary solution to this consists in looking at the set of sense pairs of a lexeme as responding to a dependence pattern. More specifically, in considering whether a sense pair corresponds to a link as being partly determined by whether there are other links already present in the same lexeme.

Take as example lexeme L, which has monolingual senses $\{s_{m1}, s_{m2}, s_{m3}, s_{m4}\}$ and bilingual senses $\{s_{b1}, s_{b2}, s_{b3}\}$, and consider the question of whether to assign a link to sense pair (s_{m4}, s_{b1}). If s_{b1} has already been linked to s_{m1}, s_{m2} and s_{m3}, and if in addition s_{m4} has already been linked to s_{b2} and s_{b3}, then it is unlikely that a further link should be added. If, on the other hand, s_{b1} has yet to be linked to any monolingual sense, and s_{m4} has yet to be linked to any bilingual sense, then it is more likely that a new link should be added. This is based on the assumption that in general we should expect each sense in one dictionary to be linked to exactly one sense in the other. Therefore we should require stronger evidence to add a second link than to add a first link, and stronger again to add a third link, etc.

The meta-classifier is designed to make use of this expectation. It applies after the ML classifier in the following manner. In a first step, it takes the confidence score p returned by the ML classifier for each sense pair (s_{mono}, s_{bil}), which it interprets as the probability of a link taking place between senses s_{mono} and s_{bil}. In total, each lexeme L gives rise to $|S_{mono}| \times |S_{bil}|$ such probability estimates. These estimates are re-calibrated using Isotonic regression, as introduced by [3], to adjust for the otherwise unnaturally low variance that arises from Adaboost averaging across all models in the ensemble.[5]

[5] The details of this are beyond the scope of the current paper, but are explained in a general way in the reference provided above.

Then, the meta-algorithm assesses whether each sense pair $(s_{mono}, s_{bil}) \in \{S_{mono} \times S_{bil}\}$ corresponds to a sense link, one at a time and in decreasing order of the estimates p output by the ML classifier. It does this by supplementing the original ML classifier estimate p with two additional probability estimates on whether the sense pair corresponds to a link. These additional estimates are computed using (a) the number of bilingual senses to which s_{mono} has already been linked, and (b) the number of monolingual senses to which s_{bil} has already been linked.

This is done invoking the cumulative distribution $\tilde{F}(x)$, which indicates the probability that a given sense is truly linked to more than x senses in the other dictionary. This function is approximated empirically as:

$$\tilde{F}(x) \approx F(x) = 1/N \times |\{s \in D \mid s \text{ has at most } x \text{ links}\}| \tag{4}$$

where $N = |D|$, that is N is the size of the whole dataset D. The values of $F(x)$ for $0 \leq x \leq 15$ are computed during the pre-processing phase, 15 being the maximum number of observed links for a sense in our dataset. These values are then stored for use by the meta-classifier in order to compute the 2 additional estimates of a link between s_{mono} and s_{bil}: (a) $1-F(m)$, and (b) $1-F(n)$, where m and n are the numbers of links already assigned to s_{mono} and s_{bil}, respectively. These estimates are then combined into a voting ensemble as:

$$(1 - \lambda_1 - \lambda_2)p_{\text{ML}}((s_{mono}, s_{bil})) + \lambda_1(1 - F(m)) + \lambda_2(1 - F(n)) \tag{5}$$

where λ_1, λ_2 are the voting weights and p_{ML} is the probability estimate of the ML classifier.

Just as the vanilla ML classifier assigns a link iff $p_{\text{ML}}((s_{mono}, s_{bil}) = 1) > 0.5$, the meta-classifier assigns a link if and only if the value in (5) is greater than 0.5. Experimentally, we found the best results setting $\lambda_1 = \lambda_2 = .25$. That is, assigning a link if and only if:

$$\begin{aligned} 0.5(p_{\text{ML}}((s_{mono}, s_{bil}))) + 0.25(1 - F(m)) + 0.25(1 - F(n)) > 0.5 \qquad &\Rightarrow \\ p_{\text{ML}}((s_{mono}, s_{bil}) = 1) > 0.5(F(m) + F(n)) & \end{aligned} \tag{6}$$

Thus, one way to view the action of the meta-classifier is as a method for varying the threshold required for linking, based on the already identified sense links in the same lexeme. The ML classifier classifies a sense pair as a link if and only if its probability estimate p_{ML} exceeds 0.5, while the meta-classifier replaces this global value (i.e., *global* in the sense that it is same for all sense pairs) with a *local* threshold T, which varies for each sense pair depending on the other sense pairs in the same lexeme.[6] Specifically,

$$T = \frac{F(m) + F(n)}{2} \tag{7}$$

If no links have yet been assigned to the senses in a sense pair (i.e., $m = n = 0$), then T will be small and so even a small probability estimate will be sufficient for a positive classification. If by contrast, the senses in question have already been linked to several other senses (i.e., m, n are large), then T will also be large and thus the estimate of the ML classifier will have to be high in order for a positive link to be assigned. The complete action of the meta-classifier is presented in Algorithm 1.

[6] To be more precise, depending on the other sense pairs with a higher p_{ML} estimate, since they will have been previously evaluated as to whether they correspond to a sense link.

Algorithm 1 Meta-classifier algorithm.

```
for each lexeme $L$ with monolingual senses set $S_{mono}$ and bilingual senses set $S_{bil}$ do
    for each sense pair $(s_{mono}, s_{bil}) \in S_{mono} \times S_{bil}$ do
        Obtain its probability estimate $p_{\text{ML}}$ from the ML classifier
    for each probability estimate $p_{\text{ML}}$, of sense pair $(s_{mono}, s_{bil})$, from largest to smallest, do
        Determine $m$ and $n$, the number of already existing links for $s_{mono}$ and $s_{bil}$, respectively
        Compute $T = \frac{F(m)+F(n)}{2}$
        if $p_{\text{ML}} > T$ then
            Classify sense pair $(s_{mono}, s_{bil})$ as a link
        else
            Classify as no link
```

Though it has only been tested on the present task of sense linking, this algorithm can in theory be generalized to any classification problem in which there is a dependency between the classification on certain sets of elements.[7]

5 Results and Discussion

As just presented, we explored two experimental settings: (a) using only a ML classifier, and (b) applying it in combination with a statistically-based meta-classifier. Table 4 provides the results for the two settings, along with those for the baseline classifier. Performance is evaluated using Precision (P), Recall (R) and its derived F1 score over sense pairs classified as *links*. Our focus was to assess the correctness of what the system had tagged as links (P on links) and its capacity to identify true links (R on links).[8] Moreover, we employed Cohen's kappa as the most common statistic used to account for correct classification that takes place purely by chance. For each metric, for each experimental setting, the best result is in bold face while the worst one is underlined.

Verbs is consistently the worst performing POS class, while the miscellaneous class *Other* performs the best in all cases but one. The good results for *Other* can be explained partly by the fact that it has a perfectly balanced training dataset (see Table 3), and partly by its low degree of polysemy (shown in Table 2), as opposed to verbs, which are the most polysemous POS. *Adverbs & Prepositions* is the second best performing class, also explained by its low degree of polysemy relative to nouns and, more particularly, verbs. Adverbs and prepositions present the highest percentage of monosemous entries, with the number of senses per entry declining very quickly. At most, 26 senses can be found in an entry for an adverb or preposition, which is half the size of the most polysemous entry for nouns.

The level of balance of each dataset (Table 3) is also a factor in the performance for each POS class. This can be appreciated when comparing P&R scores for sense pairs classified as *links* (those reported in Table 4) against P&R scores for sense pairs classified as *non-links*,

[7] For example, the task of assigning tags to YouTube videos could be viewed as a linking task between a set of videos and a set of tags. We might expect each video to be, on average, correctly assigned to around 3 tags. It would be surprising if a video was assigned no tags, or was assigned 20 tags. In the other direction, assuming the given tags were chosen so as to be meaningful and realistic, it would be surprising if one tag was not assigned any videos or if one tag was assigned to every video. These expectations could be leveraged by the meta-classifier. What is described above would require slight modification to work in other domains, but it is reasonable to conjecture that some version of the same idea may prove similarly effective elsewhere.

[8] We also obtained P and R for sense pairs classified as *non-links*, not shown here due to space constraints and given that our interest was on the correctness and coverage of *link*-tagged sense pairs.

Table 4 Performance scores for baseline, ML classifier only, and ML classifier + meta-classifier.

	Precision			**Recall**			**F1**			**Kappa**		
	base line	ML	ML+ Meta	base line	ML	ML+ Meta	base line	ML	ML+ Meta	base line	ML	ML+ Meta
Adjs	0.77	0.91	0.94	0.66	0.83	0.87	0.71	0.87	0.90	0.56	0.79	0.84
Advs-Preps	0.80	0.93	0.94	0.76	0.85	0.90	0.78	0.89	0.92	0.66	0.83	**0.88**
Nouns	0.74	0.89	0.92	0.68	0.78	0.83	0.71	0.83	0.87	0.58	0.76	0.82
Verbs	0.47	0.80	0.83	0.42	0.53	0.64	0.44	0.64	0.72	0.35	0.59	0.67
Other	**0.87**	**0.95**	**0.95**	**0.82**	**0.89**	**0.91**	**0.84**	**0.92**	**0.93**	**0.70**	**0.84**	0.87
All POS	0.70	0.88	0.91	0.63	0.75	0.80	0.66	0.81	0.85	0.54	0.74	0.80

not shown here due to space constraints. The more balanced a dataset, the more similar the P&R values for both types of sense pairs are. By contrast, in the case of verbs (the least balanced class, with only around 16% of links), the difference between the scores for *links* and *non-links* is noticeable. In the double-layer system, P and R for *non-links* respectively raise to 0.94 and 0.98 (vs. 0.83 and 0.64 for *links*). In general, the unbalance in favor of *non-links* results in high R scores for these, ranging between 0.95 and 0.98 across all POS classes. In other words, the system has a stronger tendency to identify sense pairs as *non-links*.

Overall, we assess these results as notably positive. In spite of balance issues, P and R on *links* reach a very decent level of performance. Our interest is on high P over R scores because we prefer correct links at the cost of, possibly, low coverage, which we had initially set at a minimum R score of 0.60. All POS classes attained this target. Similarly, all POS classes except for verbs reached a P score of at least 0.92, which indicates a high degree of correctness. Though not as perfect as hand-curated content, the resulting links (including those for verbs) can already be used for less quality-demanding use cases than traditional dictionaries, such as generating multilingual datasets feeding into cross-lingual search engines or image tagging systems.

Finally, Table 4 shows the positive effect of the meta-classifier. The double-layer system consistently outperforms the ML classifier alone. The improvement is most remarkable for verbs. If classified with the ML classifier only, verbs are 15 points behind *Other* in P and 36 behind in R, but the meta-classifier reduces the gap considerably, a very positive result since verbs correspond to one third of the total data (see Table 3). We thus chose the double-layer setting as our system final design.

6 Conclusions and Next Steps

This paper presented a system for linking senses between a monolingual and a bilingual dictionary. The system approaches the task as a binary classification problem, a strategy which avoids the issue of non one-to-one sense mappings between two dictionaries due to differences in sense granularity and coverage. This classifier was built using Adaboost on Decision Trees and informed with features engineered based on lexicographic knowledge.

Sense links, however, are to some extent dependent on the existence of other links for the same senses. That is, a sense in one dictionary already linked to a sense in the other has a lower probability of being linked to a further sense. Therefore, we experimented with a second classification layer to also model the dependence relation observed among sense links, which was implemented as a statistically based meta-classifier sitting on top of the ML classifier, and which resulted in significantly higher performance scores.

At this point, there are several natural next steps for this project. First, the system can already be used to generate sense links for other monolingual-bilingual dictionary pairs. Second, the double-layer system provides us with a solid framework for developing models for sense linking different types of dictionary pairs (e.g., bilingual-bilingual, monolingual-monolingual, monolingual-thesaurus, etc.), therefore contributing to the creation of a significant linguistic linked data resource. A relevant question to address as part of this work is to what extend the approach adopted here is also applicable to other dictionaries with lower degrees of curation than the ones we used. Last but not least, we can continue work towards our two-fold goal of developing methods for generating new bilingual dictionary content, as well as enriching existing ones with data from linked resources.

References

1 M. Alper. Auto-generating Bilingual Dictionaries: Results of the TIAD-2017 Shared Task Baseline Algorithm. In *Proceedings of the LDK 2017 Workshops, co-located with the 1st Conference on Language, Data and Knowledge*, pages 85–93, 2017.

2 R. J. Byrd. Discovering Relationships among Word Senses. In Antonio Zampolli, Nicoletta Calzolari, and Martha Palmer, editors, *Current Issues in Computational Linguistics: In Honour of Don Walker*, pages 177–189. Springer, Dordrecht, 1994.

3 R. Caruana and A. Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *23rd Int. Conference on Machine Learning*, pages 161–168. ACM, 2006.

4 A. Copestake, T. Briscoe, P. Vossen, A. Ageno, I. Castellón, F. Ribas, G. Rigau, H. Rodríguez, and A. Samiotou. Acquisition of lexical translation relations from MRDs. *Machine Translation*, 9:9–3, 1995.

5 K. Donandt, C. Chiarcos, and M. Ionov. Using Machine Learning for Translation Inference Across Dictionaries. In *Proceedings of the LDK 2017 Workshops*, 2017.

6 C. Fellbaum, editor. *WordNet: an Electronic Lexical Database.* MIT Press, 1998.

7 Y. Freund and R. E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.*, 55(1):119–139, August 1997.

8 T. Gollins and M. Sanderson. Improving Cross Language Retrieval with Triangulated Translation. In *24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 90–95. ACM, 2001.

9 I. Gurevych, J. Eckle-Kohler, S. Hartmann, M. Matuschek, C. M. Meyer, and C. Wirth. UBY - A large-scale unified lexical-semantic resource based on LMF. In *Proceeding of the 13th EACL Conference*, pages 580–590, 2012.

10 I. Gurevych, J. Eckle-Kohler, and M. Matuschek. *Linked Lexical Knowledge Bases: Foundations and Applications.* Morgan & Claypool Publishers, 2016.

11 N. M. Ide and J. Véronis. Mapping Dictionaries: A Spreading Activation Approach. In *Proceedings for the New OED Conference*, pages 52–64, 1990.

12 H. Kaji, S. Tamamura, and D. Erdenebat. Automatic Construction of a Japanese-Chinese Dictionary via English. In *LREC 2008*, 2008.

13 M. Lesk. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *5th Annual International Conference on Systems Documentation*, SIGDOC '86, pages 24–26, New York, NY, USA, 1986. ACM.

14 G. Massó, P. Lambert, C. Rodríguez-Penagos, and R. Saurí. Generating New LIWC Dictionaries by Triangulation. In R. E. Banchs, F. Silvestri, T. Liu, M. Zhang, S. Gao, and J. Lang, editors, *Information Retrieval Technology*, pages 263–271, 2013.

15 Mausam, S. Soderland, O. Etzioni, D. Weld, M. Skinner, and J. Bilmes. Compiling a Massive, Multilingual Dictionary via Probabilistic Inference. In *Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 262–270. ACL, 2009.

16 T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, pages 3111–3119, 2013.

17 R. Navigli and S. P. Ponzetto. BabelNet: The Automatic Construction, Evaluation and Application of a Wide-coverage Multilingual Semantic Network. *Artif. Intel.*, 193:217–250, December 2012.

18 E. Niemann and I. Gurevych. The People's Web Meets Linguistic Knowledge: Automatic Sense Alignment of Wikipedia and Wordnet. In *Ninth International Conference on Computational Semantics*, IWCS '11, pages 205–214. ACL, 2011.

19 S. P. Ponzetto and R. Navigli. Knowledge-rich Word Sense Disambiguation Rivaling Supervised Systems. In *48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1522–1531, 2010.

20 T Proisl, P Heinrich, S Evert, and B Kabashi. Translation Inference across Dictionaries via a Combination of Graph-based Methods and Co-occurrence Stats. In *LDK Workshops*, 2017.

21 M. Ruiz-Casado, E. Alfonseca, and P. Castells. Automatic Assignment of Wikipedia Encyclopedic Entries to Wordnet Synsets. In *Third International Conference on Advances in Web Intelligence*, AWIC'05, pages 380–386, 2005.

22 K. Tanaka and K. Umemura. Construction of a Bilingual Dictionary Intermediated by a Third Language. In *Proceedings of COLING'94*, pages 297–303, 1994.

23 I. Varga and S. Yokoyama. Bilingual dictionary generation for low-resourced language pairs. In *Proceedings of EMNLP*, pages 862–870, 2009. URL: `http://www.aclweb.org/anthology/D09-1090`.

24 M. Villegas, M. Melero, N. Bel, and J. Gracia. Leveraging RDF Graphs for Crossing Multiple Bilingual Dictionaries. In N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of LREC 2016*, pages 23–28, 2016.

25 Z. Wu and M. Palmer. Verbs Semantics and Lexical Selection. In *32nd Annual Meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, 1994.

26 M. Wushouer, D. Lin, T. Ishida, and K. Hirayama. Pivot-Based Bilingual Dictionary Extraction from Multiple Dictionary Resources. In *PRICAI 2014: Trends in Artificial Intelligence*, pages 221–234, Cham, 2014. Springer International Publishing.

A Features

Feature	Description
bil_dom_direct	Boolean: bilingual domain is non-empty
mono_dom_direct	Boolean: monolingual domain is non-empty
dom_col_sim_avg	co-occurrence similarity score for domain labels, avg if multiple
dom_col_sim_max	as above but max across all values if multiple
dom_col_sim_min	as above but min across all values if multiple
dom_wup_sim_avg	wu-palmer similarity score for domain labels, avg if multiple
dom_wup_sim_max	as above but max across all values if multiple
dom_wup_sim_min	as above but min across all values if multiple
bil_dom_indirect	Boolean: not all the above comparisons are non-empty
dom_cross_comps	a weighted average of the above domain-related features
bil_reg_direct	Boolean: bilingual register is non-empty
mono_reg_direct	Boolean: monolingual register is non-empty, 0 otherwise
reg_col_sim_avg	co-occurrence similarity score for register labels, avg if multiple
reg_col_sim_max	as above but max across all values if multiple

reg_col_sim_min	as above but min across all values if multiple
bil_reg_indirect	Boolean: not all the above comparisons are empty
reg_cross_comps	a weighted average of the above register-related features
bil_ge_direct	Boolean: bilingual regions is non-empty
mono_ge_direct	Boolean: monolingual region is non-empt
ge_col_sim_avg	co-occurrence similarity score for region labels, avg if multiple
ge_col_sim_max	as above but max across all values if multiple
ge_col_sim_min	as above but min across all values if multiple
bil_ge_indirect	Boolean: not all the above comparisons are empty
ge_cross_comps	a weighted average of the above region-related features
bil_ind_direct	Boolean: bilingual sense-level indicator s non-empty
ind_def_wv	cos similarity of sense-level indicators and definition, GoogleNews word vectors
ind_in_def	Boolean, word from sense-level indicators appears in definition
bil_ind_tr_direct	Boolean: bilingual translation-level indicator is non-empty
ind_tr_def_wv	cos similarity of translation-level indicators and definition, GoogleNews word vectors word vectors
ind_tr_in_def	Boolean, word from translation-level indicators appears in definition
bil_ind_ex_direct	Boolean: bilingual example-level indicator is non-empty
ind_ex_def_wv	cos similarity of example-level indicators and definition, GoogleNews word vectors word vectors
ind_ex_in_def	Boolean, word from example-level indicators appears in definition
same_number	Boolean: both marked for same countability (nouns only)
same_trans	Boolean: both marked for same transitivity (verbs only)
same_type	Boolean: both marked for same noun type, (nouns only)
all_text_comp	cos similarity of all the text from one sense with all from the other, GoogleNews word vectors
naive_bayes_estimate	see section 4.1.1
freq	comparison of frequency estimates for each sense, see section 4.1.2
mono_is_main	Boolean: monolingual sense is first in its lexeme
bil_is_main	Boolean: bilingual sense is first in its lexeme
single_sense	Boolean: this sense pair is the only sense pair in its lexeme