CogNet: a Large-Scale Cognate Database

Khuyagbaatar Batsuren[†] **Gábor Bella**[†] **Fausto Giunchiglia**[†§]
DISI, University of Trento, Trento, Italy[†]
Jilin University, Changchun, China[§]
{k.batsuren; gabor.bella; fausto.giunchiglia}@unitn.it

Abstract

This paper introduces *CogNet*, a new, large-scale lexical database that provides *cognates*—words of common origin and meaning—across languages. The database currently contains 3.1 million cognate pairs across 338 languages using 35 writing systems. The paper also describes the automated method by which cognates were computed from publicly available wordnets, with an accuracy evaluated to 94%. Finally, statistics and early insights about the cognate data are presented, hinting at a possible future exploitation of the resource[1] by various fields of lingustics.

1 Introduction

Cognates are words in different languages that share a common origin and the same meaning, such as the English *letter* and the French *lettre*. Cognates and the problem of cognate identification have been extensively studied in the fields of language typology and historical linguistics, as cognates are considered useful for researching the relatedness of languages (Bhattacharya et al., 2018). Cognates are also used in computational linguistics, e.g., for lexicon extension (Wu and Yarowsky, 2018) or to improve cross-lingual NLP tasks such as machine translation or bilingual word recognition (Kondrak et al., 2003; Tsvetkov and Dyer, 2015).

Despite the interest in using cognate data for research, state-of-the-art cognate databases have had limited practical uses from an applied perspective, for two reasons. Firstly, popular cognate-coded databases that are used in historical linguistics, such as ASJP (Wichmann et al., 2010), IELex[2], or ABVD (Greenhill et al., 2008), cover only the small set of 225 *Swadesh* basic concepts, although with an extremely wide coverage of up to 4000 languages. Secondly, in these databases, lexical entries that belong to scripts other than Latin or Cyrillic mostly appear in phonetic transcription instead of using their actual orthographies in their original scripts. These limitations prevent such resources from being used in real-world computational tasks on written language.

This paper describes *CogNet*, a new large-scale, high-precision, multilingual cognate database, as well as the method used to build it. Our main technical contributions are (1) a general method to detect cognates from multilingual lexical resources, with precision and recall parametrable according to usage needs; (2) a large-scale cognate database containing 3.1 million word pairs across 338 languages, generated with the method above; (3) *WikTra*, a multilingual transliteration dictionary and library derived from *Wiktionary* data; and (4) an online platform that lets users explore the resource.

The paper is organised as follows. Section 2 presents the state of the art. Section 3 describes the main cognate discovery algorithm and section 4 the way various forms of evidence used by the algorithm are computed. The method is parametrised and the results are evaluated in section 5. Section 6 describes the resulting *CogNet* database in terms of structure and statistical insights. Finally, section 7 concludes the paper.

2 State of the Art

To our knowledge, cognates have so far been defined and explored in two fundamental ways by two distinct research communities. On the

[1] The CogNet resource and WikTra tool are available on http://cognet.ukc.disi.unitn.it.

[2] *Indo-European Lexical Cognacy Database*, http://ielex.mpi.nl/

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3136–3145
Florence, Italy, July 28 - August 2, 2019. ©2019 Association for Computational Linguistics

one hand, *cognate identification* has been studied within linguistic typology and historical linguistics. On the other hand, computational linguists have been researching methods for *cognate production*.

The very definition of the term ‘cognate’ varies according to the research community. In historical linguistics, cognates must have a provable etymological relationship and must be translated into each language (Bhattacharya et al., 2018). Accordingly, the English *skyscraper* and the German *Wolkenkratzer* are considered as cognates but the English *song* and the Japanese ソング /songu/) are not. In computational linguistics, the notion of cognate is more relaxed with respect to etymology and loanwords are also considered as cognates (Kondrak et al., 2003). For our work we adopted the latter, computational point of view.

In historical linguistics, cognate identification methods proceed in two main steps. First, a similarity matrix of all words is estimated by three types of similarity measures: semantic similarity, phonetic similarity, and orthographic similarity. For information on semantic similarity, special-purpose multilingual dictionaries, such as the well-known *Swadesh List*, are used. For orthographic similarity, string metrics (Hauer and Kondrak, 2011; St Arnaud et al., 2017) are often employed, e.g., edit distance, Dice’s coefficient, or LCSR. As these methods do not work across scripts, they are completed by phonetic similarity, exploiting transformations and sound changes across related languages (Kondrak, 2000; Jäger, 2013; Rama et al., 2017). Phonetic similarity measures, however, require phonetic transcriptions to be *a priori* available. More recently, historical linguists have started exploiting identified cognates to infer phylogenetic relationships across languages (Rama et al., 2018; Jäger, 2018).

In computational linguistics, cognate production consists of finding for a word in a given language its cognate pair in another language. State-of-the-art methods (Beinborn et al., 2013; Sennrich et al., 2016) have employed character-based machine translation, trained from parallel corpora, to produce cognates or transliterations. (Wu and Yarowsky, 2018) also employs similar techniques, as well as multilingual dictionaries, to produce large-scale cognate clusters for Romance and Turkic languages. Although the cognates produced in this manner are, in principle, a good source for improving certain cross-lingual tasks in NLP, the quality of the output often suffers due to not being able to handle certain linguistic phenomena properly. For example, words in languages such as Arabic or Hebrew are written without vowels and machine-produced transliterations often fail to vowelize such words (Karimi et al., 2011). The solution we propose is the use of a dictionary-based transliteration tool over machine transliteration.

Our method provides new contributions for both research directions. Firstly, to our knowledge no other work on cognate generation has so far used high-quality multilingual lexical resources on a scale as large as ours, covering hundreds of languages and more than 100,000 cross-lingual concepts. Secondly, this large cross-lingual coverage could only be achieved thanks to a robust transliteration tool that is part of the contributions of our paper. Finally, our novel, combined use of multiple—orthographic, semantic, geographic, and etymological—sources of evidence for detecting cognates was crucial to obtain high-quality results, in terms of both precision and recall.

3 The Algorithm

For our work we have adopted a computational-linguistic interpretation of the notion of cognate (Kondrak et al., 2003): two words in different languages are cognates if they have the same meaning and present a similarity in orthography, resulting from a supposed underlying etymological relationship (common ancestry or borrowing).

Based on this interpretation, our algorithm is based on three main principles: (1) *semantic equivalence*, i.e., that the two words share a common meaning; (2) sufficient proof of *etymological relatedness*; and (3) the *logical transitivity* of the cognate relationship.

The core resource for obtaining cross-lingual evidence on *semantic equivalence*—i.e., the sameness of word meanings—is the *Universal Knowledge Core* (UKC), a large multilingual lexico-semantic database (Giunchiglia et al., 2018) already used both in linguistics research as well as for practical applications (Bella et al., 2016; Giunchiglia et al., 2017; Bella et al., 2017). The UKC includes the lexicons and lexico-semantic relations for 338 languages, containing 1,717,735 words and 2,512,704 language-specific word meanings. It was built from *wordnets* (Miller, 1995) and *wiktionaries* converted into wordnets (Bond and Foster, 2013)). As all of the resources composing the UKC were built and validated by humans(Giunchiglia et al., 2015), we consider the quality of our input data to be high enough for obtaining accurate results on cognates (Giunchiglia et al., 2017). As most wordnets map their units of meaning (*synsets* in WordNet terminology) to English meanings, they can effectively be interconnected into a cross-lingual lexical resource. The UKC reifies all of these mappings as supra-lingual *lexical concepts* (107,196 in total, excluding named entities such as *Ulanbaatar*). For example, if the German *Fahrrad* and the Italian *bicicletta* are mapped to the English *bicycle* then a single concept is created to which all three language-specific meanings (i.e., wordnet synsets) will be mapped.

In terms of *etymological evidence*, we use both direct and indirect evidence of etymological relatedness. Direct evidence is provided by gold-standard etymological resources, such as the one we use and present in section 4.1. Such evidence, however, is relatively sparse and would not, in itself, provide high recall. We therefore also consider indirect evidence in the form of a combined *orthographic–geographic relatedness*: a measure of geographic proximity of languages combined with the orthographic similarity of words, involving transliteration, can provide strong clues on language contact and probable cross-lingual lexical borrowing.

Finally, we exploit *logical transitivity* in order further to improve recall: we build on the intuition that if words w_a and w_b are cognates and w_b and w_c are cognates then w_a and w_c are also cognates. For example, if the German *Katze* is found to be a cognate of the English *cat* (based on direct etymological evidence) and *cat* is found to be a cognate of the French *chat* (based on orthography) then *Katze* and *chat* are also considered to be cognates).

Based on these principles, we have implemented a cognate discovery algorithm as shown in algorithm 1. Its input is a single lexical concept from the UKC (the algorithm being applicable to every concept in loop). It builds an undirected graph where each node represents a word and each edge between two nodes represents a cognate relationship.

The process starts by retrieving the lexicalisations of the input concept in all available languages and creating the corresponding word nodes in the graph (lines 2–5). All such words thus fulfil the criterion of semantic equivalence above. Then, for all different-language word pairs that express the concept (lines 6–9), we verify whether etymological evidence exists for a potential cognate relationship. The latter may either be direct evidence (*EtyRel*, line 10) or indirect, which we implement as a score of relatedness combined of orthographic similarity (*OrthSim*) and geographic proximity (*GeoProx*). We consider indirect evidence to be sufficient if this combined score is superior to an experimental threshold T_F (line 12). In case either direct or indirect evidence is found, an edge between the two word nodes is created (lines 10–13). As the last step, in order to apply the principle of logical transitivity, the *transitive closure* of the graph is computed (line 15). In the resulting graph G^+ each connected subgraph represents a group of cognate words.

Algorithm 1: Cognate Discovery Algorithm

```
Input   : $c$, a lexical concept
Input   : $\mathcal{R}$, a lexical resource
Output  : $G^+$, graph of all cognates of $c$
1  $V, E \leftarrow \emptyset$;
2  $\mathcal{L} \leftarrow \text{Languages}_{\mathcal{R}}(c)$;
3  for each language $l \in \mathcal{L}$ do
4      for each word $w \in \textit{Words}_{\mathcal{R}}(c, l)$ do
5          $V \leftarrow V \cup \{v = <w, l>\}$;
6  for each node $v_1 = <w_1, l_1> \in V$ do
7      for each node $v_2 = <w_2, l_2> \in V$ do
8          if $l_1 = l_2$ then
9              continue;
10         if $\textit{EtyRel}(w_1, l_1, w_2, l_2)$ then
11             $E \leftarrow E \cup \{e = <v_1, v_2>\}$;
12         else if $\textit{OrthSim}(w_1, l_1, w_2, l_2) + T_G \times$
             $\textit{GeoProx}(l_1, l_2) > T_F$ then
13             $E \leftarrow E \cup \{e = <v_1, v_2>\}$;
14 $G \leftarrow <$V, E$>$;
15 $G^+ = \text{TransitiveClosure}(G)$
16 return $G^+$;
```

4 Computing Etymological Relatedness

Our method predicts the etymological relatedness of words based on both direct and indirect etymological evidence. Section 4.1 below describes how the *EtyRel* function provides direct evidence. Sections 4.2 and 4.3 explain how indirect evidence is computed based on orthographic similarity using the *OrthSim* function and on geographic proximity using the *GeoProx* function.

4.1 Direct Etymological Evidence

The *EtyRel* function in algorithm 1 uses gold-standard evidence to compute the etymological relatedness of words. It exploits *etymological ancestor* (marked as *Anc* below) relations for each word of the word pair being evaluated as cognates. Two words are considered as etymologically related if they are found to have at least one common etymological ancestor word (such as the German *Ross* and the English *horse* having as ancestor the proto-Germanic root **harss-*).

$$EtyRel(w_1, l_1, w_2, l_2) = \\ = \begin{cases} \text{true} & \text{if } \mathrm{Anc}(w_1, l_1) \cap \mathrm{Anc}(w_2, l_2) \neq \emptyset \\ \text{false} & \text{otherwise} \end{cases} \tag{1}$$

Ancestor relations are retrieved from the *Etymological WordNet* (EWN)[3] (De Melo, 2014), a lexical resource providing relations between words, e.g., derivational or etymological. EWN was automatically built by harvesting etymological information encoded in *Wiktionary*. In this work, we have only used its 94,832 cross-lingual etymological relations.

4.2 Orthographic Similarity

Orthographic similarity is computed using a string similarity metric *LCSSim* based on the *longest common subsequence* (LCS) of the two input words, returning a similarity score between 0 and 1:

$$\mathrm{LCSSim}(w_1, w_2) = \frac{2 \times \mathrm{len}(\mathrm{LCS}(w_1, w_2))}{\mathrm{len}(w_1) + \mathrm{len}(w_2)} \tag{2}$$

When w_1 and w_2 belong to different writing systems, LCS returns 0 and thus the formula above is not directly usable. In order to be able to identify cognates across writing systems, we apply transliteration to the Latin script (also known as *romanization*) using the *WikTra* tool. Orthographic similarity is thus computed as:

$$\mathrm{OrthSim}(w_1, w_2) = \max\{\mathrm{LCSSim}(w_1, w_2), \\ \mathrm{LCSSim}(\mathrm{WikTra}(w_1), \mathrm{WikTra}(w_2))\} \tag{3}$$

WikTra is a dictionary-based transliteration tool compiled from information collected from *Wiktionary* and developed specifically for this work by the authors[4]. It is Unicode-based and supports 85 languages in 35 writing systems, defining transliteration rules and codes according to international standards, as developed by the Wiktionary community (the largest community in lexicography).

An illustration of the output provided by *WikTra* compared to three existing transliteration tools is provided in table 1. The use of *WikTra* with respect to existing tools is justified by a need for high-quality results that also cover complex cases of orthography, e.g., in Semitic scripts where vowels are typically omitted. In particular, *Junidecode*[5] is a character-based transliterator, an approach that seriously limits its accuracy. The *Google transliterator* is dictionary-based and is therefore of higher quality, but it supports a lower number of languages and is not freely available. Finally, *uroman* (Hermjakob et al., 2018) is a new, high-quality, dictionary-based tool that nevertheless provides a limited support for scripts without vowels (e.g., Arabic or Hebrew), as also visible in table 1.

While *WikTra* gains its high accuracy from human-curated *Wiktionary* data, it still needs to be improved for Thai and Japanese. In Thai, *WikTra* only works on monosyllabic words, and it needs an additional tool to recognize syllables. In Japanese, it only works with Hiragana and Katakana scripts and not with Kanji (Chinese characters). We therefore combined *WikTra* with the *Kuromoji*[6] transliteration tool.

4.3 Geographic Proximity

We exploit geographic information on languages in order to take into account the proximity of language speakers for the prediction of borrowing. Our hypothesis is that, even if in the last century lexical borrowing on a global scale has been faster than ever before, the effect of geographic distance is still a significant factor when applying cognate discovery to entire vocabularies. This effect is combined with orthographic similarity in line 12 of algorithm 1, in a way that geographic proximity increases the overall likelihood of word pairs being cognates, without being a necessary condition.

[3] http://www1.icsi.berkeley.edu/∼demelo/etymwn/, accessed on 10/14/2018.

[4] https://github.com/kbatsuren/wiktra

[5] https://github.com/gcardone/junidecode

[6] https://github.com/atilika/kuromoji

Table 1: Comparison with state-of-the art transliteration tools

#	Languages	Word	Uroman	Junidecode	Google	WikTra
1	English	book	book	book	book	book
2	Malayalam	മലയാളം	malayaallam	mlyaallN	malayāḷaṁ	malayāḷaṃ
3	Arabic	نواة	nwaa	nw@	nawa	nawātun
4	Japanese	コンピュータ	konpyuta	konpiyuta	konpyūtā	konpyūtā[*]
5	Thai	ราชาทิรา	raachaatiraa	raachaathiraad	rā chā thi rād	raa-chaa-tí-râat [b]
6	Russian	москва	moskva	moskva	moskva	moskva
7	Hindi	देवनागरी	devanaa	devnaagrii	devanaagaree	devnāgrī
8	Bengali	বাংলা	baangla	baaNlaa	bānlā	bangla
9	Greek	ἀναϋ̄τέ	anaute	anauteo	anāftéo	anautéō
10	Kashmiri	کَمپیُوٹَر	kampivwuttar	khampy[?]w?ttar	-	kampeūṭar
11	Persian	ارمنستان	armnstan	rmnstn	-	armanestân
12	Hebrew	יששכר	yshshkr	yshshkr	yissachar	yiśśāḵār
13	Tamil	ரெஃஸ்	rehs	reHs	reḥs	rex
14	Ethiopic	አዲስ አበባ	aadise aababaa	‘aadise ’aababaa	ādīsi ābeba	ʾädis ʾäbäba
15	Tibetan	ཁ་པར	kha·pa	kh-pr	-	kha par
16	Korean	메가폰	megapon	megapon	megapon	megapon
17	Armenian	Հայաստան	hayiastan	hayastan	hayastan	hayastan
18	Uyghur	ئائىله	yeayealae	y’y’-lae	-	a’ile
19	Khmer	ក្រមារ	kromaaro	krmaar	krama r	krâméar
20	Telugu	అంకపాళి	amkapali	aNkpaalli	aṅkapāḷi	aṅkapāḷi
21	Odia	ଓଡ଼ିଶା	oddishaa	rodd’ishaa	-	oriśa
22	Burmese	သည်းခြေ	sannykhre	snny[?]:kh[?]e	saeehkyay	sany:hkre

[*] WikTra in Japanese language only works with scripts of Hiragana and Katakana.
[b] WikTra in Thai language only works with a sequence of syllables.

Our relatively simple solution considers only the languages of the input words, computing a language proximity value between 0 and 1, as follows:

$$\text{GeoProx}(l_1, l_2) = \min\left(\frac{T_D}{\text{GeoDist}(l_1, l_2)}, 1.0\right) \tag{4}$$

The function GeoDist(l_1, l_2) is an approximate ‘geographic distance’ between two languages l_1 and l_2, based on the geographical areas where the languages are spoken. The constant T_D corresponds to a *minimal distance*: if two languages are spoken within this distance then they have maximum geographic relatedness. T_D is empirically set as described in section 5.2.

Distances between languages are provided by the WALS resource[7], one of most comprehensive language databases. WALS provides latitude and longitude coordinates for a language given as input. While a single coordinate returned for a language may in some cases be a crude approximation of linguistic coverage (e.g., Spanish is spoken both in Spain and in most countries of Latin America), even this level of precision was found to improve our evaluation results.

5 Evaluation

This section describes how CogNet was evaluated on a manually built cognate corpus and how its parameters were tuned to optimise results.

5.1 Dataset Annotation

While our initial idea was to use existing cognate datasets for evaluation, the most comprehensive databases turned out to represent cognates in their phonetic transcriptions instead of having words written in their original scripts. Such data was not usable to test our method that performs transliteration on its own.

Consequently, we created a dataset of 40 concepts with fully annotated sets of cognate groups. On average, a concept was represented in 107 languages by 129 words: 5,142 words in total for the 40 concepts. The concepts were chosen from the *Swadesh basic word list* and from the WordNet *core concepts* (Boyd-Graber et al., 2006). The lexicalizations (words) corresponding to these concepts were retrieved from the UKC. For each concept, we asked two language experts to find cognate clusters among its lexicalizations. The experts made their decisions based on online resources such as Wiktionary and the *Online Etymology Dictionary*[8]. Cohen’s Kappa score, inter-

[7] https://wals.info

[8] https://www.etymonline.com

Table 2: Parameter configuration and comparisons.

Methods	T_F	T_G	T_D	P	R	F_1
Baseline 1: LCS	0.60	-	-	94.70	25.62	40,32
Baseline 2: Consonant	-	-	-	98.07	19.11	31,98
LCS + Geo	0.60	0.01	1.3	94.02	27.63	42.71
LCS + Geo + EWN	0.60	0.01	1.3	94.10	30.41	45.97
LCS + Geo + WikTra	0.63	0.02	1.2	94.15	42.42	58.49
LCS + Geo + WikTra + EWN	0.63	0.02	1.2	94.20	44.86	60.78
LCS + Geo + Trans	0.68	0.02	1.2	95.94	44.27	60.59
LCS + Geo + Trans + EWN	0.70	0.06	1.3	97.32	53.53	69.07
LCS + Geo + Trans + WikTra	0.72	0.06	1.2	94.14	77.59	85.07
LCS + Geo + Trans + WikTra + EWN	0.71	0.04	1.1	93.94	86.32	89.97

annotator agreement, was 95.15%. The resulting human-annotated dataset contained 5,142 words, 38,447 pairs of cognate words and 320,338 pairs of non-cognate words. We divided this dataset into two equal parts: the first 20 concepts for parameter configuration and the second 20 concepts for evaluation.

5.2 Algorithm Configuration

The goal of configuration was to optimise the algorithm with respect to three hyperparameters: the threshold of combined orthographic–geographic relatedness T_F (section 3), the geographic proximity contribution parameter T_G, and the minimum distance T_D (section 4.3).

We have created a three-dimensional grid with $T_F = [0.0; 1.0]$ (the higher the value, the more the strings need to be similar to be considered as cognates), $T_G = [0.0; 1.0]$ (the higher the value, the more geographic proximity is considered as evidence), and $T_D = [0.0; 22.0]$ (here, the unit of 1.0 corresponds to a distance of 1000km, within which geographic relatedness is a constant maximum).

In this grid, we computed optimal values for each parameter (in increments of 0.01) based on performance on the configuration dataset described in section 5.1. With these optimal settings, we evaluated all possible combinations of the various components of the cognate generation method, in order to understand their relative contribution to the overall score. Since our ultimate goal is to generate high-quality knowledge, we favoured precision over recall, setting our minimum precision threshold to 95% and maximizing recall with respect to this constraint. The best settings (computed on the parameter configuration dataset) as well as the corresponding precision–recall figures (computed on the evaluation dataset) are reported in table 2. Although we set the precision threshold to 95% for the configuration dataset, we obtained precision results that are slightly lower, about 94%, on the evaluation dataset.

The results of configuration can be seen in table 2. The optimal *geographic region* parameter T_D varies between 1.1 and 1.3, which correspond to a radius of 1,100–1,300km: languages spoken within such a distance tend to share more cognates.

One interesting insight from table 2 concerns the use of logical transitivity. While it is an extremely efficient component in our algorithm, in order to maintain precision it requires the relatedness threshold T_S to be increased from [0.6; 0.63] to [0.68; 0.71] and the influence of geographic relatedness T_G from [0.1; 0.2] to [0.2; 0.6]. This means that in order for transitivity to hold, both the overall relatedness criterion and the geographic proximity need to become more strict.

5.3 Evaluation Results

We evaluated the effect of the various components of our method (geographic relatedness, WikTra transliteration, Etymological WordNet, transitivity) on its overall performance. As a baseline, we used two string similarity methods often used in cognate identification (St Arnaud et al., 2017): *LCS*, i.e., the longest common subsequence ratio of two words (which we also use in equation 2), and *Consonant*, which is a heuristic method that checks if the first two consonants of the words are identical. Although the baseline *Consonant* method achieved the highest precision of 98.07%, its recall is the lowest, 19.11%, due to being limited to Latin characters.

Adding geographic proximity, direct etymological evidence, and transliteration to the algorithm increased recall in a consistent manner, by about 2%, 3%, and 15%, respectively, all the while maintaining precision at the same level. Computing the transitive closure, finally, had a major multiplicator effect on recall, bringing it to 86.32%. With this full setup we were able to generate 3,167,642 cognate pairs across 338 languages.

In order to cross-check the quality of the output, we randomly sampled 400 cognate pairs not covered by the evaluation corpus and had them re-evaluated by the same experts. Accuracy was found to fall in the 93–97% range, very much in line with the goal of 95% we initially set in section 5.2.

6 Exploring CogNet

At an accuracy of 94%, our algorithm has generated 3,167,642 cognates. They cover 567,960 words and 80,836 concepts, corresponding to 33.06% of all words and 73.52% of all concepts in the UKC: one word out of three and three concepts out of four have at least one cognate relationship.

In terms of WordNet formalism, cognate relationships can be expressed as *cross-lingual sense relations* that connect (*word*, *synset*) pairs—reified in wordnets as senses—across languages. As not all wordnets represent senses explicitly, CogNet encodes these relationships in the following tuple form:

$$(\textit{PWN\_synset}, w_1, l_1, w_2, l_2, \textit{metadata})$$

where *PWN_synset* is the Princeton WordNet English synset ID representing the shared meaning of the cognate pair, w_1 and w_2 are the two words, l_1 and l_2 are their respective languages (expressed as ISO-639-3 codes), and *metadata* is a set of attributes describing the cognate pair, such as the type of evidence for the relationship (direct etymological or indirect). The entire CogNet resource is described and freely downloadable from the web[9].

While we expect CogNet to provide linguistic insights for both theoretical and applied research, we are just starting to exploit its richness. As a first result, we have developed an online tool[10] for the visual exploration of cognate data (see figure 1 for an illustration). In the long term, this web tool is intended for linguists both for the exploration of data and for collaborative work on extending the resource.

We also carried out an initial exploration of cognate data along the axes of *language*, *language family*, and *geographic distance*. Figure 2 shows the number of cognates found at a given geographic distance (i.e., the distance of the speakers of the two languages, as defined in section 4.3). We observe that the vast majority of cognates is found within a distance of about 3,000km. Our interpretation of these results is that, by and large, locality is still a major influence on modern lexicons, despite the globalising effects of the last centuries. Let us note that the geographic proximity component of our algorithm alone could not have caused this distribution, as it had a relatively minor overall contribution on the results (see the geographic factor $T_G = 0.04$ in table 2).

In order to avoid biasing per-language statistics by the incompleteness of the lexicons (wordnets) used, we limited our study to the 45 languages with a vocabulary size larger than 10,000 words. As a further abstraction from lexicon size, we introduce the notion of *cognate density*, defined over a set of words as the ratio of words covered by at least one cognate pair of CogNet. In other words, working with cognate densities allows us to characterise the 'cognate content' of each language independently of the wordnet size.

Cognate densities for the 45 languages studied show a wide spread between languages with the highest density (the top five language being Indonesian: 60.80%, Czech: 59.05%, Catalan: 58.66%, Malay: 57.63%, and French: 57.25%) and those with the lowest (the bottom five languages being Thai: 7.87%, Arabic: 9.01%, Persian: 9.64%, Mongolian: 10.37%, and Mandarin Chinese: 11.03%). The main factor behind high cognate density is the presence of closely related languages in our data: as Malay and Indonesian are mutually intelligible registers of the same language, the existence of separate wordnets for the two naturally results in a high proportion of shared vocabulary. Inversely, languages on the other end of the spectrum tend not to have major living languages that are closely related. Let us finally note that non-perfect transliteration and failed transliteration-based matches may also be a reason for low cognate recall for languages with very different scripts, such as Chinese, Arabic, or

[9] http://cognet.ukc.disi.unitn.it

[10] http://linguarena.eu

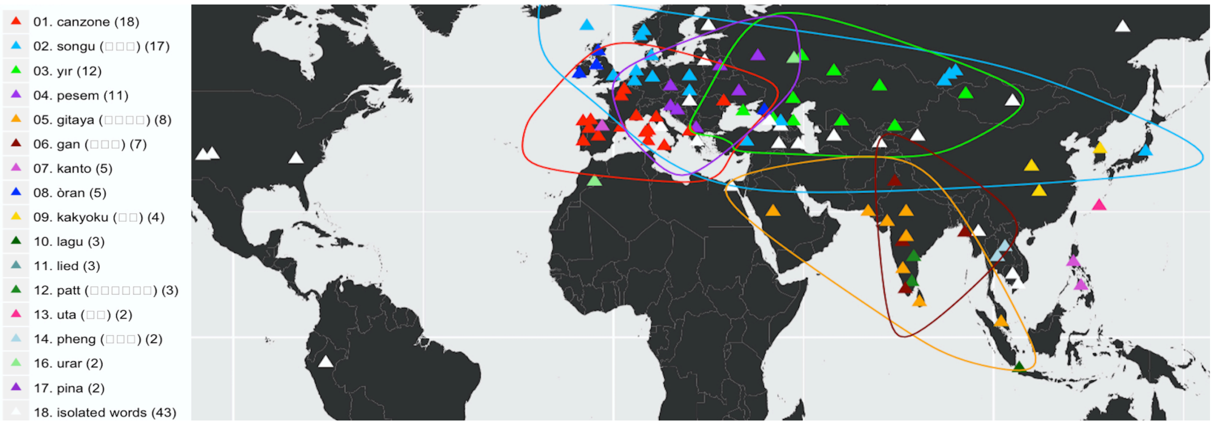

Figure 1: Cognate sets of the concept ‘song’, represented with different colours. It is easy to observe the effects of language families (e.g., red triangles stand for Romance languages) and geographic proximity (e.g., the higher density of orange in South-West Asia and green in Central Asia).

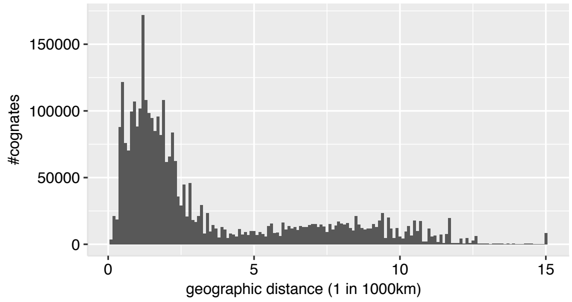

Figure 2: The number of cognates according to the geographic distance of the language speakers.

Family	Density	Family	Density
Malay	59.22%	Greek	22.99%
Romance	53.32%	Niger-Congo	18.63%
Slavic	36.67%	Japanese	12.16%
Indo-Aryan	36.08%	Sino-Tibetan	11.22%
Germanic	34.10%	Mongolian	10.37%
Basque	32.82%	Persian	9.64%
Dravidian	24.79%	Arabic	9.01%
Finno-Ugric	24.57%	Thai	7.87%

Table 3: Cognate density by language family, computed over the 45 largest-vocabulary languages.

Thai.

In order to verify these intuitions, we examined cognate densities for the 45 languages manually clustered into 16 language families (see table 3, the language name was kept for clusters of size 1). Indeed, families such as Malay, Romance, Slavic, or Indo-Aryan, well known for containing several mutually intelligible language pairs, came out on top, while families with generally fewer or mutually non-intelligible members at the bottom. The only outlier is Basque that, despite being an isolate, is close to the resource-wide average cognate density of 33%.

7 Conclusions

In this paper, we have demonstrated a general method for building a cognate database using existing wordnet resources. Identifying cognates based on orthography for words written in 35 different writing systems, as opposed to phonetic data, made the problem statement novel with respect to existing research in cognate identification. The use of a large-scale cross-lingual database and a combination of linguistic, semantic, etymological, and geographic evidence resulted in what in our knowledge is the largest cognate database both in terms of the number of concepts and of the writing systems covered. The evaluation showed that the resource has promisingly high quality, with precision and recall adjustable through the algorithm parameters. The resource has been made available online, together with a graphical web-based tool for the exploration of cognate data, our hope being to attract both linguists and computer scientists as potential users.

Acknowledgments

This paper was partly supported by the *InteropEHRate* project, co-funded by the European Union (EU) Horizon 2020 programme under grant number 826106.

The first author is supported by the Cyprus Center for Algorithmic Transparency, which has received funding from the European Union’s Horizon 2020 Research and Innovation Program under Grant Agreement No. 810105.

References

Lisa Beinborn, Torsten Zesch, and Iryna Gurevych. 2013. Cognate production using character-based machine translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 883–891.

Gabor Bella, Fausto Giunchiglia, and Fiona McNeill. 2017. Language and Domain Aware Lightweight Ontology Matching. *Web Semantics: Science, Services and Agents on the World Wide Web*.

Gabor Bella, Alessio Zamboni, and Fausto Giunchiglia. 2016. Domain-Based Sense Disambiguation in Multilingual Structured Data. In *The Diversity Workshop at the 22nd European Conference on Artificial Intelligence (ECAI 2016)*.

Tanmoy Bhattacharya, Nancy Retzlaff, Damián E Blasi, William Croft, Michael Cysouw, Daniel Hruschka, Ian Maddieson, Lydia Müller, Eric Smith, Peter F Stadler, et al. 2018. Studying language evolution in the age of big data. *Journal of Language Evolution*.

Francis Bond and Ryan Foster. 2013. Linking and extending an open multilingual wordnet. In *ACL (1)*, pages 1352–1362.

Jordan Boyd-Graber, Christiane Fellbaum, Daniel Osherson, and Robert Schapire. 2006. Adding dense, weighted connections to wordnet. In *Proceedings of the third international WordNet conference*, pages 29–36. Citeseer.

Gerard De Melo. 2014. Etymological wordnet: Tracing the history of words. In *LREC*, pages 1148–1154. Citeseer.

Fausto Giunchiglia, Khuyagbaatar Batsuren, and Gabor Bella. 2017. Understanding and exploiting language diversity. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, pages 4009–4017.

Fausto Giunchiglia, Khuyagbaatar Batsuren, and Abed Alhakim Freihat. 2018. One world—seven thousand languages. In *Proceedings 19th International Conference on Computational Linguistics and Intelligent Text Processing, CiCling2018, 18-24 March 2018*.

Fausto Giunchiglia, Mladjan Jovanovic, Mercedes Huertas-Migueláñez, and Khuyagbaatar Batsuren. 2015. Crowdsourcing a large scale multilingual lexico-semantic resource. In *AAAI Conference on Human Computation and Crowdsourcing (HCOMP-15)*.

Simon J Greenhill, Robert Blust, and Russell D Gray. 2008. The austronesian basic vocabulary database: from bioinformatics to lexomics. *Evolutionary Bioinformatics*, 4:EBO–S893.

Bradley Hauer and Grzegorz Kondrak. 2011. Clustering semantically equivalent words into cognate sets in multilingual lists. In *Proceedings of 5th international joint conference on natural language processing*, pages 865–873.

Ulf Hermjakob, Jonathan May, and Kevin Knight. 2018. Out-of-the-box universal romanization tool uroman. *Proceedings of ACL 2018, System Demonstrations*, pages 13–18.

Gerhard Jäger. 2013. Phylogenetic inference from word lists using weighted alignment with empirically determined weights. *Language Dynamics and Change*, 3(2):245–291.

Gerhard Jäger. 2018. Global-scale phylogenetic linguistic inference from lexical resources. *CoRR*, abs/1802.06079.

Sarvnaz Karimi, Falk Scholer, and Andrew Turpin. 2011. Machine transliteration survey. *ACM Computing Surveys (CSUR)*, 43(3):17.

Grzegorz Kondrak. 2000. A new algorithm for the alignment of phonetic sequences. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 288–295. Association for Computational Linguistics.

Grzegorz Kondrak, Daniel Marcu, and Kevin Knight. 2003. Cognates can improve statistical translation models. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume of the Proceedings of HLT-NAACL 2003–short papers-Volume 2*, pages 46–48. Association for Computational Linguistics.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Taraka Rama, Johann-Mattis List, Johannes Wahle, and Gerhard Jäger. 2018. Are automatic methods for cognate detection good enough for phylogenetic reconstruction in historical linguistics? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 2, pages 393–400.

Taraka Rama, Johannes Wahle, Pavel Sofroniev, and Gerhard Jäger. 2017. Fast and unsupervised methods for multilingual cognate clustering. *arXiv preprint arXiv:1702.04938*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1715–1725.

Adam St Arnaud, David Beck, and Grzegorz Kondrak. 2017. Identifying cognate sets across dictionaries of related languages. In *Proceedings of the EMNLP 2017*, pages 2519–2528.

Yulia Tsvetkov and Chris Dyer. 2015. Lexicon stratification for translating out-of-vocabulary words. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 125–131.

Søren Wichmann, André Müller, Viveka Velupillai, Cecil H Brown, Eric W Holman, Pamela Brown, Sebastian Sauppe, Oleg Belyaev, Matthias Urban, Zarina Molochieva, et al. 2010. The asjp database (version 13). *URL: http://email. eva. mpg. de/˜ wichmann/ASJPHomePage. htm*, 3.

Winston Wu and David Yarowsky. 2018. Creating large-scale multilingual cognate tables. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018).*