PBML

The Prague Bulletin of Mathematical Linguistics

NUMBER 114 APRIL 2020

EDITORIAL BOARD

Published twice a year by Charles University (Prague, Czech Republic)

Editorial office and subscription inquiries:
ÚFAL MFF UK, Malostranské náměstí 25, 118 00, Prague 1, Czech Republic
E-mail: pbml@ufal.mff.cuni.cz

ISSN 0032-6585

PBML

The Prague Bulletin of Mathematical Linguistics

NUMBER 114 APRIL 2020

CONTENTS

Articles

PBML

The Prague Bulletin of Mathematical Linguistics

NUMBER 114 APRIL 2020 5–33

ParaDis and Démonette – From Theory to Resources for Derivational Paradigms

Fiammetta Namer,[a] Nabil Hathout[b]

[a] UMR 7118, ATILF, Université de Lorraine & CNRS, France
[b] UMR 5263, CLLE, Université de Toulouse Jean-Jaurès & CNRS, France

Abstract

In this article, we trace the genesis of the French derivational database Démonette and show how its architecture and content stem from recent theoretical developments in derivational morphology and from user needs. The development of this large-scale resource began a year ago as part of the Demonext project. Its conception is grounded in a theoretical approach where the lexemes are connected by derivational relations within derivational families which in turn fit into paradigms. More precisely, Démonette is a partial implementation of ParaDis, a paradigmatic model of morphological representation designed for the description of regular processes and of form-meaning discrepancies. The article focuses on the principles that govern the morphological, structural and semantic encoding of morphologically complex lexemes in Démonette and illustrates the range of form-meaning discrepancies with a variety of examples of non-canonical word formations.

1. Introduction

Démonette is a large-scale derivational database of French developed as part of the Demonext project funded by the French National Research Agency (ANR-17-CE23-0005). The project started in 2018. Its main goal is the description of 366,000 morphological relations covering a large range of processes that includes conversion, prefixation and suffixation. About 120 affixes are used in these derivations. They include suffixes like *-ard*, *-ariat*, *-at*, *-âtre*, *-el*, *-aie*, *-iser*, *-erie*, *-esque*, *-esse*, *-eur*, *-eux*, *-iste*, and prefixes like *a-*, *anti-*, *bi-*, *co-*, *contre-*, *dé-*, *é-*, *extra-*, *hyper-*, *hypo-*, *in-*, *infra-*, *inter-*. Démonette's descriptions primarily come from existing reliable resources created and

Corresponding author: fiammetta.namer@univ-lorraine.fr

Cite as: Fiammetta Namer, Nabil Hathout. ParaDis and Démonette – From Theory to Resources for Derivational Paradigms. The Prague Bulletin of Mathematical Linguistics No. 114, 2020, pp. 5–33.
doi: 10.14712/00326585.001.

distributed as part of various academic works such as PhD projects. The data extracted from these databases is reanalysed in order to fit in the Démonette format and complemented when some features are missing. New entries extracted from machine readable dictionaries will also be added.

Démonette's entries are morphological relations between two lexemes (L_1, L_2) described by morphological, formal, categorical and semantic features. The database is highly redundant by design in order to be flexible enough and to have the capability to represent the many non-canonical morphological relations that occur in Word Formation (WF) of many languages. It is based on the theoretical principles that govern ParaDis, a paradigmatic model of derivational morphology where the multidimensional structure of the lexeme is generalized. ParaDis is based on two fundamental structures: derivational families (networks of lexemes) and paradigms (aligned sets of families). The article details the motivation of these theoretical assumptions and explains how they are implemented.

The remainder of the paper is organized as follows. In Section 2, we present an overview of some existing derivational databases (DDBs) that were created for different European languages. The theoretical background and motivation of the paradigmatic approach adopted in Démonette are then discussed in Section 3 with a focus on ParaDis and on the representation of various non-canonical derivational phenomena in this model. In Section 4, we first detail the structure of Démonette and the way the morphological, categorical and semantic features are encoded, and then show the capability of the proposed formalism to represent a number of non-canonical derivations that occur in French.

2. Resources and tools in morphology

Morphological analysis is part of the initial pre-processing task in many natural language processing (NLP) systems. The morphological analyzers they use are often based on machine learning and statistical methods. Words are decomposed into morphemes in order to compensate for the limitations of lexicons. Let us mention systems like Linguistica (Goldsmith, 2001), Morfessor (Creutz and Lagus, 2005), or, more recently, Cotterell and Schütze (2017)'s models. These systems may be used for any language provided that enough training data is available, however they are more effective for concatenative morphology languages such as English, German and French. Morphological analysis may also be carried out by symbolic (rule-based) systems developed by linguists; for a panorama, see (Bernhard et al., 2011).

Morphological parsers can be replaced or supplemented in the NLP pipeline by large enough lexical resources containing derivational annotations if their features are sufficiently rich and varied. However, very few such resources exist for most languages[1]. One of the firsts is CELEX (Baayen et al., 1995), which describes the phonetic, inflectional, morpho-syntactic, derivational and statistical properties of 216,775 lemmas of Dutch, English and German. The entries are extracted from dictionaries and corpora (news and literature). A more recent English DDB is CatVar (Habash and Dorr, 2003) which includes 100,000 lexemes grouped in subfamilies. A similar resource is DerivBase (Zeller et al., 2013) which describes 215,000 German lexical entries gathered in semantically motivated derivational families. Two other DDBs have recently been developed for German from CELEX, namely DErivCelex (Shafaei et al., 2017) and Morphological Treebank (Steiner and Ruppenhofer, 2018). Several DDBs have also been created for Slavic languages like CroDeriv (Šojat et al., 2014) for Croatian and DeriNet (Žabokrtský et al., 2016; Ševčíková et al., 2017; Vidra et al., 2019) for Czech. DeriNet is a lexical network that captures core word-formation relations connecting around 970,000 Czech lexemes. Derivational relations between verbs and some of their nominal derivatives are described in version 3.0 of Princeton (English) WordNet (Fellbaum et al., 2009) which provides a semantic characterization of the noun with respect to the verb. *Employer*$_\mathrm{N}$ is for instance described as the "agent" of *employ*$_\mathrm{V}$. DDBs have also been developed for Romance languages, mainly French, Italian and Latin. DerIvaTario is a derivational dictionary of Italian (Talamo et al., 2016) which provides descriptions based on strong hypotheses regarding allomorphy and suppletion. For instance, *bellicoso* 'bellicose' is analyzed as a derivative of *guerra* 'war'. Word Formation Latin (WFL) (Litta et al., 2016) is a derivational morphology resource for Classical Latin, where the lemmas (i.e., the non-inflected words) are decomposed into their formative components, and relations between the lemmas are identified by Word Formation Rules (WFRs). WFL contains 69,682 lemmas.

Few resources also exist for French. The JeuxDeMots platform (Lafourcade and Joubert, 2008), a serious game, has created a large coverage lexical network where the words are connected by semantic relations. These relations are inspired by the lexical function formalism (Mel'čuk, 1996). Some of them are derivational. JeuxDeMots being a crowd-sourced resource, the accuracy of the related pairs of words proposed by the players increases with the number of identical answers. In 10 years, the size of this resource has reached 270 million relations (pairs of words) that instantiate 150 different lexical functions. It connects 3.5 million words and expressions.

However, French still lacks true large-scale resources primarily aimed at the description of derivational morphology. To fill this gap, we developed a prototype database called Démonette$_\mathrm{V1}$ (Hathout and Namer, 2014b, 2016) from 2011 to 2017. Démonette$_\mathrm{V1}$ describes 73,233 derivational families made up of a verb, its agent and action noun derivatives and its modality adjective. Three objectives were pursued: (*i*)

[1] In his exhaustive review, Kyjánek (2018) proposes a typology of the structures and coverage of 30 derivational resources for Romance (including Latin), Germanic and Slavic languages and provides a complete list of the main existing DDBs and resources that contain derivational annotations. The reader may refer to this report which is far more complete than the present overview.

use the DériF morphological analyzer (Namer, 2009, 2013) to produce a resource made up of derivational relations between pairs of lexemes L_1 and L_2, labelled with linguistically grounded features, including semantic annotations; (*ii*) complement these $L_1 \rightarrow L_2$ derivations by indirect relations between members of the same derivational family extracted from the Morphonette lexicon (Hathout, 2009); (*iii*) define an extensible and redundant architecture which can be fed by heterogeneous morphological resources.

The design of the current Démonette database (Section 4) is based on the experience gained during the development of Démonette$_{V1}$. The aim of the second version is to produce a resource which provides descriptions (morphological, phonological, categorial, distributional, and especially semantic) that may be useful for NLP, but also serve as a reference for several audiences including research in morphology, teaching, language and speech therapy practice. The structure of the database must be flexible enough to allow for a (semi-)automatic acquisition of morphological descriptions from existing resources. It must also be able to include any non canonical formation or any additional derivation (affixation, conversion and even composition). To this purpose, the architecture of the database is based on theoretical principles that ensure a uniform representation of regular derivation (words where meaning and form mirror each other) and non-canonical derivation which infringe form-meaning compositionality. Démonette implements the principles of ParaDis, a model which borrows from lexeme-based and paradigm-based approaches to WF (Section 3).

3. Démonette's theoretical background

Démonette is based on two fundamental principles: (*i*) the adoption of the lexeme as unit of analysis, and (*ii*) the organisation of the morphological lexicon into paradigms. These principles have independently contributed to recent evolution of morphological theories, and influenced the content and organization of many derivational resources. This section briefly describes how these major modifications came about.

3.1. Morphemes and the form-meaning non-compositionality

The morpheme, conceived as the minimal bi-faced unit of meaning and form, is the descriptive and analytic unit morphologists have used in the so-called morpheme-based morphological traditions, whether concatenative (Item and Arrangement) or functional (Item and Process) (Hockett, 1954). Morpheme-based approaches have been adopted in many morphological analyzers and resources, including CELEX (Baayen et al., 1995), DerIvaTario (Talamo et al., 2016), CroDeriV (Šojat et al., 2014), the Morphological Treebank (Steiner and Ruppenhofer, 2018), and the first version of Word Formation Latin (Litta et al., 2016).

The main advantage of morpheme-based approaches is their simplicity and their capacity to describe all morphologically complex words by means of a small set of minimal units. Their integration into broader NLP systems is therefore very easy. In this approach, morphological rules handle only one type of unit, morphemes; they yield head-argument structures similar in nature to the outputs of syntactic rewriting rules: affixes are heads that select either lexical roots, or combinations of morphemes produced by other rules. The consequence of this similarity is that syntactic and morphological analysis and generation can be performed by a uniform grammatical system that operates on a reduced lexicon only made up of morphemes. However, this efficiency comes at a high cost. Morpheme-based morphology suffers from well-known limitations that have been widely discussed in the literature (among others, see (Aronoff, 1976; Anderson, 1992; Fradin, 2003); a recent, in-depth review is given in (Blevins, 2016)). The most significant drawback is the rigidity of the morpheme because it requires all form to be associated with a meaning and vice-versa. With such a strong constraint, the analysis for non canonical derivation processes (Corbett, 2010) becomes far too complex. Morphemes also prove unfit for the description of non-concatenative morphology languages such as templatic morphology in Semitic languages, tonal or stress shifting systems, etc. This takes away all interest in morpheme. Table 1 illustrates some of these limitations with English, French and Italian examples; similar formations exist in many other European languages including Spanish and German.

	WF	Lang.	Lexeme$_1$	Lexeme$_2$
a.	conversion	eng	*nurse*$_{\mathrm{N}}$	*nurse*$_{\mathrm{V}}$
b.	parasynthesis	fra	*banque*$_{\mathrm{N}}$ 'bank'	*interbancaire*$_{\mathrm{A}}$ 'between banks'
c.	parasynthesis	eng	*departement*$_{\mathrm{N}}$	*interdepartemental*$_{\mathrm{A}}$
d.	affix replacement	eng	*fascism*$_{\mathrm{N}}$	*fascist*$_{\mathrm{N}}$
e.	polysemy	fra	*porter*$_{\mathrm{V}}$ 'carry'	*porteur*$_{\mathrm{Nm,[hum]\ or\ [artif]}}$ 'carrier'
f.	synonymy	ita	*compatto*$_{\mathrm{A}}$ 'compact'	*scompattare*$_{\mathrm{V}}$ or *decompattare*$_{\mathrm{V}}$ 'uncompact'
g.	back-formation	eng	*vivisect*$_{\mathrm{V}}$ 'perform vivisection'	*vivisection*$_{\mathrm{N}}$ 'vivisection'

Table 1. Examples of meaning-form discrepancies in English, French and Italian derivational relations

The first example in Table 1(a) is a *zero affixation* or *conversion* (Tribout, 2012), also often called zero-morpheme (Dahl and Fábregas, 2018); it is characterized by Hathout and Namer (2014a) as a case of "formal under-marking" of the derivative with respect to its base since the form of the verb is identical to the form of the noun whereas the semantic content of the verb is more complex: the verb *nurse* can be paraphrased as “act as a nurse”, which includes the semantic content of the nominal homonym “nurse”; the predicative meaning “acting as” has no formal realization.

Another well-known type of meaning-form asymmetry illustrated in Table 1(b, c) is the so-called *parasynthetic* derivation; for recent overviews, see (Hathout and Namer, 2018; Iacobini, 2020). For instance, in Table 1(c), the *-al* suffix does not contribute to the meaning of *interdepartmental*, which only combines the meaning of the noun *department* and of the prefix *inter-*: “between departments”. Likewise, in Table 1(b), the *-aire* suffix does not intervene in the construction the semantic content of *interbancaire* “between banks” which is derived by *inter-* prefixation, from the semantic content of the noun *banque*. At first sight, the prefix *inter-* is responsible of the semantic operation (“between Xs”) and the suffix *-aire* of the change of categories (N→A). However, this analysis is challenged by the existence of unsuffixed adjectives like *interbank*, *interbirth*, *intercategory*, *interdeparment*, *interfamily* in English or *interbanques*, *interdépartements*, *intercellules*, *interatomes* in French. For instance, *interdepartment communication* can be found in English, or *transactions interbanques* “interbank transactions” in French. In other words, in Table 1(b) and 1(c), the prefix *inter-* assigns the derived words to their semantic class and grammatical category, whereas the suffix plays no role in the construction. Therefore, these derivatives are considered as “over-marked” in Hathout and Namer (2014a) because one of their formal elements does not have any semantic or categorial contribution.

Table 1(g) illustrates a similar case. On the semantic level, the verb could be considered as derived from the noun, since it has a more complex content than the noun. On the other hand, *vivisection* is 10 times more frequent than *vivisect* (10 times more Google hits) and its first occurrence is older: according to the Oxford dictionary, for example, the noun was in use at the beginning of the 18th century, whereas the verb’s first occurrence dates back to mid 19th century. Moreover, *vivisect* means “perform a vivisection”. In other words, *vivisect* is under-marked twice: the additional meaning in *vivisect* does not have a formal counterpart and there is no meaning associated with the *-ion* suffix in *vivisection*. This so-called “affix substraction” (Manova, 2011) is also known as *back-formation* (Becker, 1994).

The derivational relation between pairs of lexemes like *fascism/fascist* in Table 1(d) is analyzed as an *affix replacement* (Booij and Masini, 2015): Lexeme$_2$ is coined by replacing the *-ism* suffix in Lexeme$_1$ by *-ist* and vice-versa. Therefore, the two lexemes are “under- and over-marked” with respect to one another.

Other non-canonical derivations involve processes that produce two series of words that have the same form but different meanings or with different forms but the same meaning. In the first case (absence of formal markdown), the derivatives are *polysemous* as in Table 1(e) where French *-eur* suffixed nouns can denote humans and artifacts. In the second case, the derivatives are *synonymous*. This *morphological variation* results from a rivalry or *competition* between derivational processes which apply to the same base. In the Italian example in Table 1(f), the prefixes *s-* and *de-* compete to form deadjectival verbs (Todaro, 2017). When applied to the same adjective *compatto*, they produce two synonymous verbs, *scompattare* and *decompattare*. This absence of semantic markdown can be regarded as a derivational equivalent of Thornton (2012)'s notion of *overabundance*.

3.2. Lexemes, and non-binary or non-oriented rules

The shift from morpheme to *lexeme* solves several problems that arise from meaning non-compositionality. Unlike the morpheme, the lexeme is not a concrete minimal unit. It is actually an abstract object (i.e. a noninflected word, in the simplest cases) that records the common properties of the inflectional paradigm it stands for, in the form of an autonomous three-dimensional structure: (*i*) a set of phonological form (or stems); (*ii*) a part-of-speech; (*iii*) a meaning. In this framework, *word* (or *lexeme*) *formation rules* (WFRs) are oriented relations between two schemata. Each of these schemata specifies the constraints the lexemes must meet in order to enter the relation and to activate the WFR. For instance, the English WFRs in the first column of Table 2 derive relation adjectives from nouns by suffixation in *-al* (Table 2(1a)) and *-ic* (Table 2(1b)); Table 3(1) presents the English WFR that converts nouns to similative verbs. The WFR states that the input nouns must denote human beings and that the output verbs are transitive. The derivational relations *government*/*governmental* in Table 2(2a), *atom*/*atomic* in Table 2(2b) and *nurse*/*nurse* in Table 3(2) respectively instantiate the WFR in Table 2(1a), Table 2(1b) and Table 3(1).

Because WFRs apply independently and simultaneously to all three levels of description (formal, categorial and semantic), more than one formal exponent can be associated with one semantic type of derivatives (like the competing affixes *-al* and *-ic* in Table 2). Similarly, a category-shifting process can be realized without any formal change as in Table 3. Conversely, one formal exponent can be associated with more than one semantic category of derivatives: for example, denominal adjectives of material, like *wooden* and deadjectival causative verbs like *blacken* are suffixed with the same *-en* exponent.

As we said, many problems illustrated in Table 1 are solved by the shift from morpheme to lexeme. A conversion like in Table 1(a) simply modifies the semantic content and the part-of-speech but leaves the formal content unchanged as shown in Table 3; similarly, a polysemous affixation as in Table 1(e) involves two distinct WFRs, one for humans and the other for artifacts. However, these WFRs are identical on the formal level: they use the same formal exponent to derive different semantic contents. Conversely, synonymy (Table 1(f)) corresponds to cases where two (or more than two) different WFRs apply to the same input lexeme (e.g. *compatto*) and produce two (or more than two) different formal realizations associated with the same derived semantic content.

However, some problems remain because WFRs are abstractions of oriented relations designed to connect derived words to their bases. They are for instance unfit for

	(1) WFR	(2) Example
(a)	$\begin{bmatrix}\text{/X/}\\ \text{N}\\ \text{‘@’}\end{bmatrix} \rightarrow \begin{bmatrix}\text{/Xl/}\\ \text{A}\\ \text{‘pertaining to @’}\end{bmatrix}$	$\begin{bmatrix}\text{/ˈɡʌvənmənt/}\\ \text{N}\\ \text{‘government’}\end{bmatrix} \rightarrow \begin{bmatrix}\text{/ˌɡʌvənˈməntl/}\\ \text{A}\\ \text{‘pertaining to}\\ \text{the government’}\end{bmatrix}$
(b)	$\begin{bmatrix}\text{/X/}\\ \text{N}\\ \text{‘@’}\end{bmatrix} \rightarrow \begin{bmatrix}\text{/Xɪk/}\\ \text{A}\\ \text{‘pertaining to @’}\end{bmatrix}$	$\begin{bmatrix}\text{/ˈætəm/}\\ \text{N}\\ \text{‘atom’}\end{bmatrix} \rightarrow \begin{bmatrix}\text{/əˈtɒmɪk/}\\ \text{A}\\ \text{‘pertaining to}\\ \text{the atom’}\end{bmatrix}$

*Table 2. Two N→A*suf *Word Formation Rules in English*

(1) WFR	(2) Example
$\begin{bmatrix}\text{/X/}\\ \text{N}\\ \text{‘@}_{\text{+hum}}\text{’}\end{bmatrix} \rightarrow \begin{bmatrix}\text{/X/}\\ \text{V}_{\text{+transitive}}\\ \text{‘act as a @’}\end{bmatrix}$	$\begin{bmatrix}\text{/nɜːs/}\\ \text{N}\\ \text{‘nurse’}\end{bmatrix} \rightarrow \begin{bmatrix}\text{/nɜːs/}\\ \text{V}\\ \text{‘act as a nurse’}\end{bmatrix}$

Table 3. N→V Word Formation Rule in English

the description of non-oriented and indirect derivational relations like affix replacement (Table 1(d)). Likewise, back-formation (Table 1(g)) cannot be represented by means of WFRs because the formal and semantic parts of the relation have opposite orientations, nor are they able to describe parasynthetic derivation like in (Table 1(b, c)). In this case, the limitation does not result from the orientation of the WFRs, but from the fact that these derivatives are produced by a ternary WF device. More specifically, classical WFRs cannot predict the value of the supernumerary suffix mark nor explain the diversity of these suffixes, as illustrated in Table 4. The adjectives *interbanc**aire***, *intercellul**aire***, *interocéan**ique***, *interethn**ique***, *intertrib**al*** or *interparoissi**al*** all describe a spatial interval between two or more concrete entities (‘between several X’) where the noun X is, respectively, *banque*, *cellule*, *océan*, *ethnie*, *tribu* and *paroisse*. Prefixation in *inter-* may therefore involve at least three different suffix values (*-aire*, *-ique* and *-al*) but this value cannot be deduced from the form nor the meaning of the base. In other words, the adjectives in Table 4 cannot be properly analyzed without an access to the set of all the lexemes derivationally related to the base noun, as we will see below.

X_N	**between several X_N**	X_N	**between several X_N**
banque 'bank'	*interbancaire*	*cellule* 'cell'	*intercellulaire*
océan 'ocean'	*interocéanique*	*ethnie* 'ethny'	*interethnique*
tribu 'tribe'	*intertribal*	*paroisse* 'parish'	*interparoissial*

Table 4. Examples of the (X, interXsuf) noun-to-adjective relation in French

3.3. Paradigms, and partially motivated relations

Derivational paradigms solve most of the above-mentioned limitations raised by the lexeme-based approaches (for a panorama, see Štekauer (2014)) because paradigmatic relations are not necessarily binary nor are they oriented (base → derivative). In a paradigmatic framework (Bonami and Strnadová, 2019), the central unit is the *derivational family*, i.e. a structured set of lexemes[2] whose forms and meanings depend on each other. More specifically, all the members of a derivational family are interconnected just like all the inflected forms of a lexeme. Figure 1 adapted from (Bonami and Strnadová, 2019) presents a paradigm of four families made up of a verb and three derivatives (e.g. *advertise*, *advertiser*, *advertisement*, *advertisee*).

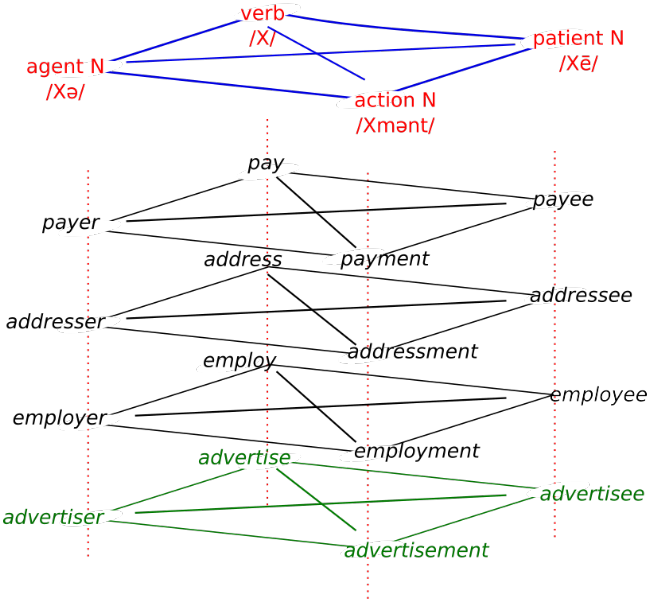


Figure 1. Regular paradigm

[2] However, the notions of paradigm and of lexeme are independent.

Two families F_1 and F_2 belong to the same paradigm when they line up so that members of the same rank or position in F_1 and F_2 are in the same form and meaning relations with the other members of their family. The aligned lexemes belong to the same *derivational series* (Hathout, 2011). For instance, the nouns *payee*, *addressee*, *employee*, *advertisee* in Figure 1 form a derivational series. A derivational paradigm can then be defined as a set of aligned families as illustrated in the lower part of Figure 1 which could be seen as a concrete paradigm in the sense of (McCarthy and Prince, 1993). The corresponding abstract paradigm is given in the upper part of Figure 1. An abstract paradigm is a network connecting the descriptions of the four derivational series.

A paradigm may contain some incomplete families, that is, families where some lexemes are missing with respect to other more complete families. Aligned incomplete families form sub-paradigms. For example, the paradigm in Figure 1 could be complemented by the 3-members family (*refuse*, *refuser*, *refusee*) which belongs to the sub-paradigm (verb-/X/, agent_N-/Xə/, patient_N-/Xiː/), but where the *-ment* action noun is missing.

Paradigm-based frameworks present two major advantages: flexibility and completeness. They are flexible because they do not only consider oriented base→derivative relations, and complete because their fundamental units are the derivational families. These two properties enable paradigm-based models to take into account affix replacement (Table 1(d)) and back-formation (Table1(g)) in a straightforward way.

In a paradigmatic approach of WF, the relations between the members of a derivational family are all represented in the same way, as non-oriented schemata, be the relations direct (base→derivative) or not. For example, the schema (1) describes the relation between *fascist* and *fascism* of (Table 1(d)): the "@1" and "@2" variables stand for the semantic content of *fascist* and *fascism* respectively, and X stands for the sequence /fæʃ/ they have in common. The mutual motivation of the two nouns can be expressed by a cross-definition of their semantic content: *fascism* is defined as the "ideology defended by a fascist" and *fascism* as a "follower of fascism"[3]. The *fascism*↔*fascist* pair is a partial family that fits in a larger paradigm represented in Table 5. The triplets (Table 5(a, b)) connect a noun or a proper name referring to an entity X, a noun of ideology (Xism) that values that entity, and a human noun (Xist) denoting a person supporting that ideology. Bochner (1993) represents these paradigmatic relations in the theoretical framework of the *Cumulative Patterns* as a ternary schema as in (2)[4].

[3] Booij and Masini (2015) propose a slighly different way to formalize cross-formation patterns by means of so-called "second order schemata".

[4] Other theoretical frameworks have been proposed to represent paradigms in derivation by Koenig (1999); Booij (2010); Spencer (2013); Antoniova and Štekauer (2015) to only cite a few.

	X: valued entity	Xism: ideology	Xist: follower
a.	*Calvin*	*calvinism*	*calvinist*
b.	*race*	*racism*	*racist*
c.	–	*fascism*	*fascist*

Table 5. (X, Xist, Xism) paradigm in English

(1) $$\begin{bmatrix} \text{/Xɪst/} \\ \text{N} \\ \text{@1:‘follower of @2’} \end{bmatrix} \leftrightarrow \begin{bmatrix} \text{/Xɪzm/} \\ \text{N} \\ \text{@2: ‘ideology defended by @1’} \end{bmatrix}$$

(2) $$\left\{ \begin{bmatrix} \text{/Xɪst/} \\ \text{N} \\ \text{@1:‘follower of @2,} \\ \text{endorsing @3’} \end{bmatrix}, \begin{bmatrix} \text{/Xɪzm/} \\ \text{N} \\ \text{@2: ‘ideology} \\ \text{defended by @1,} \\ \text{promoting @3’} \end{bmatrix}, \begin{bmatrix} \text{/X/} \\ \text{PrN or N} \\ \text{@3: ‘entity} \\ \text{promoted by @2,} \\ \text{endorsed by @1’} \end{bmatrix} \right\}$$

The description of back-formation (Table 1(g)) is also straightforward, because the paradigmatic relation (3) is not oriented. On the formal level, it connects *vivisect* to *vivisection* where X stands for /ˌvɪvɪˈsek/. On the semantic level, it connects *vivisection* to *vivisect* through the definition of ‘@1’(the meaning of the verb) as being derived from ‘@2’ (the meaning of the dynamic noun).

(3) $$\begin{bmatrix} \text{/Xt/} \\ \text{V} \\ \text{@1:‘perform @2’} \end{bmatrix} \leftrightarrow \begin{bmatrix} \text{/Xʃən/} \\ \text{N}_{\text{+dyn}} \\ \text{@2} \end{bmatrix}$$

Nevertheless, the questions raised by the parasynthetic derivations in Table 1(b,c) still remain to be answered. They are illustrated in Table 6 which repeats and extends Table 4 where we have seen that all the prefixed adjectives contain a suffix, that its value is variable and that it cannot be predicted from the form nor meaning of the noun; we have remarked that the suffix does not contribute to the meaning of the prefixed adjective, which is derived directly from the meaning of the noun. For all these reasons, we have said that the prefixed adjective is over-marked with respect to its base noun.

However, the over-marking is not arbitrary, as illustrated in Table 6: the suffix that appears in the prefixed adjective (column 3) is always the same as the suffix of the relational adjective (column 2) from the noun (column 1). In other words, the form of the prefixed adjective is derived from the relational adjective of the noun while its meaning is derived from the meaning of the noun. Their construction uses both the semantic properties of X_{N} and the formal properties of $\mathrm{X}\mathit{suf}_{\mathrm{A}}$. Therefore, the *inter*$\mathrm{X}\mathit{suf}_{\mathrm{A}}$ adjectives have two bases, one semantic (the noun X_{N}) and one formal (the relational adjective $\mathrm{X}\mathit{suf}_{\mathrm{A}}$).

X_N	X*suf*$_A$: 'of X'	*inter*X*suf*$_A$: 'between several Xs'
banque	*bancaire*	*interbancaire*
cellule	*cellulaire*	*intercellulaire*
tribu	*tribal*	*intertribal*
paroisse	*paroissial*	*interparoissial*
océan	*océanique*	*interocéanique*
ethnie	*ethnique*	*interethnique*
corail 'coral'	*corallien*	*intercorallien*
bactérie 'bactery'	*bactérien*	*interbactérien*

*Table 6. (*X, Xsuf, interXsuf*) paradigm in French.*

The classical paradigmatic approaches mentioned above consider that WF takes place in the derivational families. However, they are not able to handle the parasynthetic derivatives because they are designed to describe derivational relations where the three dimensions of the lexeme (form, category and meaning) co-vary. To overcome this limitation, we need a model where the semantic and formal relations are described separately, as they are in ParaDis.

3.4. ParaDis

Asymmetric formations like the ones in Table 1(b, c) are far from exceptional. They occur in French and in many European languages and concern a large portion of the denominal prefixed adjectives. In French, these adjectives describe spatial relations (*inter-*, *intra-*, *sous-*, *sur-*, etc.), temporal relations (*pré-*, *post-*, *anté-*, etc.), opposition (*anti-*), support (*pro-*), quantification (*mono-*, *bi-*, *pluri-*, etc.) and many others. They also concern denominal verbs like *lieu* 'place' → *localiser* 'localize'; for a full overview, see (Hathout and Namer, 2014a).

In order to account for these formations, we need a model that transposes the main contribution of lexeme-based morphology (the independent formal, categorial and semantic levels of representations) to the paradigmatic organization of the lexicon. The model must combine a morpho-phonological structure where the form of an *inter*X*suf*$_A$ adjective is connected to the form of the corresponding X*suf*$_A$ with a morpho-semantic structure where the meaning of *inter*X*suf*$_A$ is related to the meaning of X_N.

This description can be framed in the theoretical framework ParaDis "Paradigms vs Discrepancies" (Hathout and Namer, 2018) which generalize the three levels structure of the lexicon to the derivational paradigms. Our assumption is that derivational morphology is paradigmatic because the morpho-semantic regularities, the morpho-categorial regularities and the morpho-formal regularities are paradigmatic.

In other words, the organization of the derivational paradigms is extended to the semantic, categorial and formal levels of representation. In ParaDis, a morphological paradigm is an abstract combination of a morpho-formal paradigm, a morpho-categorial paradigm and a morpho-semantic paradigm just as a lexeme is the abstract combination of a formal, a categorial and a semantic descriptions. The combination of the three paradigms is obtained by mapping each of them to the morphological paradigm. In this framework, the formal, categorial and semantic levels are independent in the sense that there are not directly connected. This independence is key to the description of the asymmetric formations like *interbancaire*, as shown in Figure 2. For the sake of readability, the categorial and the semantic levels have been merged in the remainder of this article.

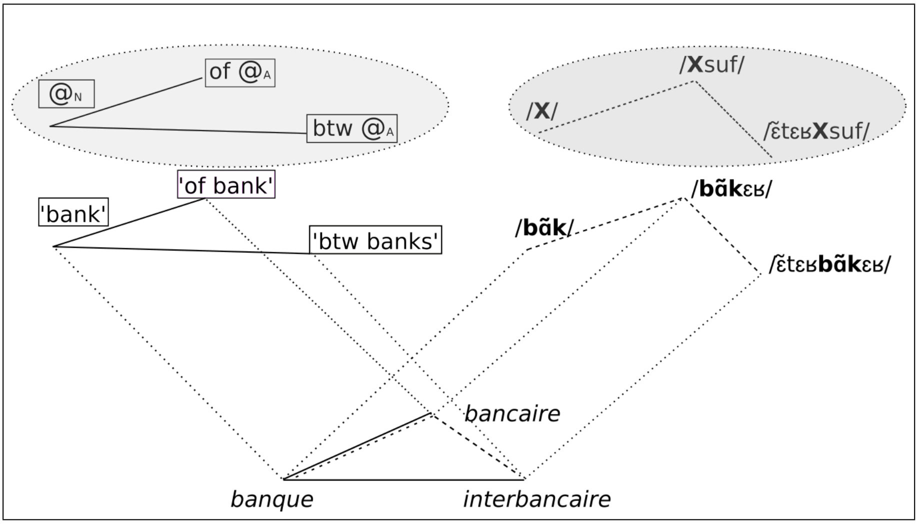


Figure 2. ParaDis. Representation of the (X_N, $X\text{suf}_A$, interXsuf_A) *asymmetrical paradigm*

In Figure 2, the gray oval on the right represents a formal abstract paradigm defined as a network of formal series. This abstract paradigm is an abstraction of a concrete formal paradigm defined as an alignment of formal families. It is represented in the figure by a single formal family (i.e. the network that is just below), the one that contains the form of the prefixed adjective *interbancaire*. The other formal families of the concrete formal paradigm are omitted. In the formal family, the vertices are phonological representations and the edges (dashed lines) describe the formal motivation that hold between these representations.

Similarly, the gray oval on the left represents a semantic abstract paradigm and the network just below the semantic family of the meaning of the prefixed adjective. The nodes in this graph represent meanings (morpho-semantic values) and the edges (solid lines) describe how they are related to the other meanings contained in the family.

The graphs of the formal and the semantic families are incomplete and different. In the semantic graph, the meanings 'btw banks' and 'of banks' cannot be deductible one from the other; these nodes are not connected. Likewise, /bɑ̃k/ and /ɛ̃tɛʁbɑ̃kɛʁ/ are not connected in the formal family because the ending (i.e. suffix) of the latter cannot be predicted from the former.

The formal and semantic concrete paradigms are in correspondence (dotted lines) with a morphological paradigm represented in the lower part of the figure by one of the morphological families it contains, namely the family of *interbancaire*. As mentioned above, morphological paradigms are alignments of morphological families and morphological families are connected graphs of lexemes.

When applied to the (*banque*, *bancaire*, *interbancaire*) family, the projection of the formal and semantic families on the morphological family results in three types of relations. The relation between *banque* and *bancaire* is regular (two lines, one solid and one dashed): it inherits the semantic motivation from the semantic family, and the formal motivation from the formal family. On the other hand, the relation between *bancaire* and *interbancaire* has only a formal motivation (dashed line) and the relation between *banque* and *interbancaire* has only a semantic motivation (solid line). The other families of Table 6 are analyzed in the same way.

X_N	$\mathrm{X}\mathit{suf}_\mathrm{A}$	$\mathit{pluri}\mathrm{X}\mathit{suf}_\mathrm{A}$	$\mathit{multi}\mathrm{X}\mathit{suf}_\mathrm{A}$
'@'	'of @'	'with more than one @'	
atome 'atom'	*atomique*	*pluriatomique*	*multiatomique*
cellule 'cell'	*cellulaire*	*pluricellulaire*	*multicellulaire*
clone 'clone'	*clonal*	*pluriclonal*	*multiclonal*
os 'bone'	*osseux*	*pluriosseux*	*multiosseux*

Table 7. (X_N, Xsuf_A, $\mathrm{pluriXsuf}_A$, $\mathrm{multiXsuf}_A$) families in French

The above analysis can be extended to the French families of the form (X_N, $\mathrm{X}\mathit{suf}_\mathrm{A}$, $\mathit{pluri}\mathrm{X}\mathit{suf}_\mathrm{A}$, $\mathit{multi}\mathrm{X}\mathit{suf}_\mathrm{A}$) illustrated in Table 7. In these families both prefixes express plurality; *multicellulaire* and *pluricellulaire* are synonymous; they mean 'with more than one cell'. These families raise two meaning-form issues. First, the prefixed adjectives (in columns 3 and 4) are always over-marked, and contain a semantically neutral suffix borrowed from the same relational adjective in column 2, whereas their meaning is directly computed from the semantic content '@' of the noun X in column 1. The different values of the suffix (*-ique, -aire, -al, -eux*) reflect the competition that exists between these WF processes. The second issue is that the two prefixed adjectives are synonymous: with concrete nominal bases, the *pluri-* and *multi-* prefixes can be freely substituted one for the other (Amiot, 2005). Similar synonymous parasynthetic adjectives also exist in other Romance languages and in English. The combination of these mismatches results in the apparent irregularity of the derivational relations within the families in Table 7. The way they are analyzed in ParaDis is illustrated by Figure 3.

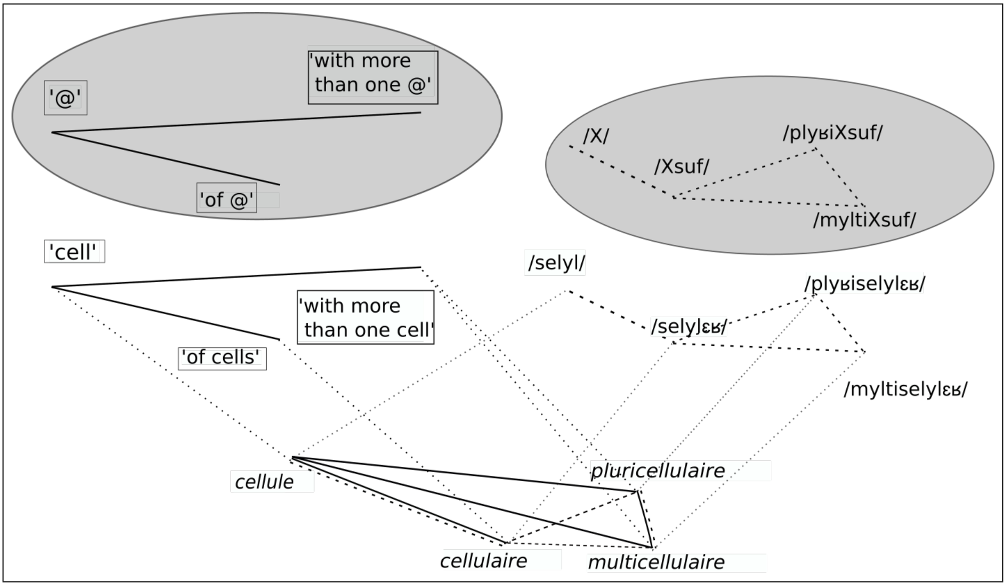


Figure 3. ParaDis. Representation of the X_N, $Xsuf_A$, $pluriXsuf_A$, $multiXsuf_A$*) asymmetrical paradigm*

Semantically, the families in Table 7 form a three-cells paradigm similar to the semantic paradigm in Figure 2: the entity '@' is connected to the relation 'of @' and the modifier 'with more than one @'. In this graph, the meanings of the relation and the modifier are not directly related. On the formal side (at the top right), /selylɛʁ/, /plyʁiselylɛʁ/ and /myltiselylɛʁ/ are connected because they are inter-predictible; /selylɛʁ/ depends on /selyl/; on the other hand, /plyʁiselylɛʁ/ and /myltiselylɛʁ/ cannot be predicted from /selyl/. The meaning-form asymmetry in the morphological families in Table 7 results from the differences between the formal and the semantic graphs. The two graphs have different sizes with four vertices for the formal network but only three for the semantic one. Because the semantic graph is smaller, the difference in size expresses a regular synonymy: the meaning 'with more than one cell' corresponds to two distinct members in the morphological family (*multicellulaire* and *pluricellulaire*).

The next section shows how the main features of ParaDis are implemented in Démonette.

4. The Démonette derivational database

Démonette is a derivational database fully compatible with the principles presented in Section 3. It is able to uniformly represent the classical binary and oriented derivation processes and all the meaning-form mismatches illustrated in Table 1.

Démonette has an original structure: its entries are *derivational relations* between pairs of lexemes that belong to the same family. They are not limited to relations between a base and one of its derivatives and include back-formations, cross-formations, parasynthetical derivatives, etc. In addition to the initial 96,000 entries of Démonette$_{V1}$ (Hathout and Namer, 2014b; Namer et al., 2017), Démonette is fed by several existing derivational resources developed and validated by morphologists. These reliable resources contain detailed semantic and phonological descriptions. 183,000 entries will be added in this way. Most of them are direct relations corresponding to ca. 120 derivational processes: conversion; suffixation in *-ard*, *-ariat*, *-at*, *-âtre*, *-el*, *-aie*, *-iser*, *-erie*, *-esque*, *-esse*, *-eur*, *-eux*, *-iste*, etc.; prefixation in *a-*, *anti-*, *bi-*, *co-*, *contre-*, *dé-*, *é-*, *extra-*, *hyper-*, *hypo-*, *in-*, *infra-*, *inter-*, etc. The original base→derivative relations are cast into the Démonette's format and new information, new pairs and new lexemes are (semi-)automatically added when necessary.

In what follows, we present Démonette's general structure (§ 4.1). We then detail how the regular and irregular derivational relations are represented (§ 4.2), including polysemy, conversion, back-formation and cross-formation (see Table 1). We also show how synonymous and parasynthetic derivatives are represented in Démonette (§ 4.3).

4.1. Overview

Démonette implements the main features of ParaDis. Its structure is based on the following principles, some of which having been already implemented in Démonette$_{V1}$:

- a *record* or *entry* describes a relation between two lexemes that belong to a derivational family; is identified by a pair of lexemes;
- a *lexeme* that takes part in several relations will be described in as many records;
- an entry (L_1, L_2) contains the *description* of L_1, of L_2 and of relation that holds between them (that is, the derivational pattern that generalizes the relation (L_1, L_2));

L	PoS	Inflectional paradigm (Latinate root)	Ontological type	...
planter	V	plɑ̃t, plɑ̃tɔ̃, plɑ̃te, plɑ̃tɛ, ... (plɑ̃tat)	Dynamic Situation	...
planteuse	Nfem	plɑ̃tøz	Person|Artifact	...

Table 8. Démonette. Excerpt of the table of lexemes

- some *features* of a lexeme are independent of the relations it appears in. They include the standardized written form of the lexeme, its part-of-speech, its inflectional paradigm (in IPA format), a possible set of learned roots (e.g. the latinate root plɑ̃tat for the verb *planter* 'plant'; see Table 8), and its ontological type selected among the 25 WordNet Unique Beginners (UB) (Miller et al., 1990). The description of the lexemes are grouped into a **table of lexemes**. Table 8 presents an excerpt of the records of the verb *planter*$_{\mathrm{V}}$ 'plant' and *planteuse*$_{\mathrm{Nfem}}$ 'female planter' or 'instrument used to plant (trees)'. Derivational polysemy is described in the table of lexemes where the ambiguity between several related meanings are indicated by the symbol "|", meaning "or", as illustrated by the ontological type of *planteuse*;
- *relations* between lexemes are stored in a separate table, the **table of relations**;
- a *relation* $(\mathrm{L}_1, \mathrm{L}_2)$ is defined by three independent sets of features: morphological (characterization of the morphological process connecting L_1 to L_2), formal (description of the formal variation between L_1 and L_2) and semantic (semantic category of the relation and glosses define L_1 and L_2 with respect to one another).

The remainder of the paper details the architecture of Démonette and focuses on the formal, structural and semantic features of the table of relations; readers can refer to (Namer et al., 2017) for a presentation of the morpho-phonological features. We also show how meaning-form discrepancies are taken into account and how this resource can provide a large-scale description of the paradigmatic organization of the morphologically complex lexicon.

4.2. (Almost) regular paradigms in Démonette

Démonette is a suitable tool for the representation of regular relations in word-formation: canonical derivation, cross- and back-formation, conversion, cf. Table 1. First, consider the relations between the lexemes of the families in Table 9. Theses families contain a verb (the predicate *laver* 'wash'), the corresponding iterative verb (the predicate *relaver* 're-wash'), the action noun derived from the two predicates (*lavage* 'washing' and *relavage* 'rewashing'), and an adjective expressing potentiality (*lavable* 'washable'). In French, action nouns may be derived by conversion (*découper* → *découpe*) or suffixation in *-age*, *-ment*, *-ion*, *-ure*, etc.; all these processes are in competi-

	X_V	$X(suf)_N$	reX_V	$reX(suf)_N$	$Xable_A$
a.	*laver* 'wash'	*lav**age***	*relaver*	*relav**age***	*lavable*
b.	*classer* 'rank'	*classe**ment***	*reclasser*	*reclasse**ment***	*classable*
c.	*planter* 'plant'	*plantat**ion***	*replanter*	*replantat**ion***	*plantable*
d.	*souder* 'weld'	*soud**ure***	*resouder*	*resoud**ure***	*soudable*
e.	*découper* 'cut (out)'	*découpe*	*redécouper*	*redécoupe*	*découpable*

Table 9. (X_V, $X(suf)_N$, reX_V, $reX(suf)_N$, $Xable_A$) families in French

tion. However, in each families in Table 9, the action nouns of the two predicates are always derived by the same formal process. All the derivational relations between the members of the families in Table 9 are regular because they all are formally and semantically motivated. They form complete oriented graphs. These graphs contain 20 edges and each of them is an entry in Démonette. We will see in § 4.2.1 and § 4.2.2 how the formal and the semantic features interact in order to represent different categories of regular WF relations illustrated by the family of *laver* (Table 9(a)).

4.2.1. Morphological features

Table 10 lists the morphological relations that hold between the members of the family of *laver* with their structural and morphological features. The relations in the other families in Table 9 are described in the same way; the formal aspect related to the conversion in Table 9(e) are discussed in Table 11. The morphological description of a relation (L_1, L_2) involves five features: `Orientation` and `Complexity` encode the structural properties of the relation; the values of `Schema`$_{L1}$ and `Schema`$_{L2}$ correspond to the morphological patterns of L_1 and $L_2$with respect to this relation; `Morph(ological) Match(ing)` combines `Schema`$_{L1}$ and `Schema`$_{L2}$.

- The `Orientation` of entry (L_1, L_2) indicates whether L_1 is an ancestor of L_2 (`a2d`; ancestor to descendant), whether L_2 is an ancestor of L_1 (`d2a`; descendant to ancestor) or whether the relation is `indirect`.
- `Complexity` describes the number of morphological steps needed to reach L_2 from L_1. When one lexeme is the base of the other, the value is `simple` (e.g. *laver* is the base of *lavage*). The value is also `simple` when L_1 and L_2 have a common base (e.g. *lavage* and *relaver* are both derived from *laver*). Notice that a derived word may have more than one base. For example, *relavage* is derived from *relaver* by suffixation in *-age* and from *lavage* by prefixation in *re-*. In both entries (*relavage*, *relaver*) and (*relavage*, *lavage*), `Orientation` is `a2d` and `Complexity` is `simple`. The value `complex` is used in all the other cases. A `complex` relation has a `a2d` or `d2a` orientation when it connects an ancestor and a descendant and involves at least two steps (e.g. (*relavage*, *laver*) is a two-steps relation where *laver*

L_1	L_2	Schema_{L1}	Schema_{L2}	Morph Match	Orientation	Complexity
laver	*lavage*	X	Xage	X/Xage	a2d	simple
laver	*relavage*	X	reXage	X/reXage	a2d	complex
laver	*relaver*	X	reX	X/reX	a2d	simple
laver	*lavable*	X	Xable	X/Xable	a2d	simple
lavage	*relavage*	X	reX	X/reX	a2d	simple
lavage	*relaver*	Xage	reX	Xage/reX	indirect	simple
lavage	*lavable*	Xage	Xable	Xage/Xable	indirect	simple
relaver	*relavage*	X	Xage	X/Xage	d2a	simple
relavage	*lavable*	reXage	Xable	reXage/Xable	indirect	complex
relaver	*lavable*	reX	Xable	reX/Xable	indirect	simple

Table 10. Démonette. Structural and formal features of the morphological relations that hold in the family of laver

is an ancestor of *relavage*). The relations are `complex, indirect` if they hold between two distant members and neither of them is a descendant or an ancestor of the other, e.g. (*relavage*, *lavable*).

- $\texttt{Schema}_{L1}$ and $\texttt{Schema}_{L2}$ describe the exponents of L_1 to L_2 in the relation that connects L_1 to L_2: X represents the sequence they have in common in this context. Therefore, the schemata are relation-dependent and vary with respect to the relation. For instance, *relavage* is annotated `Xage` with respect to *relaver* and `reXage` with respect to *lavable*.
- `Morph(ological) Match(ing)` is a concatenation of the values of $\texttt{Schema}_{L1}$ and $\texttt{Schema}_{L2}$. Two relations with identical `Morph(ological) Match(ing)` belong to the same morphological series regardless of the part of speech involved: the (*laver*$_V$, *lavage*$_{Nmas}$) and (*relaver*$_V$, *relavage*$_{Nmas}$) pairs share the value `X/Xage` and therefore belong to the same series; likewise, (*laver*$_V$, *relaver*$_V$) and (*lavage*$_{Nmas}$, *relavage*$_{Nmas}$) belong to the same series, identified by the `X/reX` value.

For the sake of space, the relations in Tables 10, 11 and 12 are listed in only one direction. Any entry (L_1, L_2) in Démonette has a symmetrical entry (L_2, L_1). In this entry, the values of $\texttt{Schema}_{L1}$, $\texttt{Schema}_{L2}$ are inverted; the value of `Morpho(logical) Match(ing)` is the mirror of that of (L_1, L_2); the value `a2d` becomes `d2a` and vice-versa for the feature `Orientation`; the other features are unchanged.

Verb-noun conversion (Table 11) can be described with the same five features. Since conversion does not involve exponents, the values of $\texttt{Schema}_{L1}$ and $\texttt{Schema}_{L2}$ are always `X` and `X` and the value of `Morphological Matching` is always `X/X`. The lack of exponent makes it impossible to decide which of the noun and the verb is the base; for a complete analysis of verb-noun conversion in French, see (Tribout, 2012). Therefore the `Orientation` is non-documented (`nd`) for the conversions, as in Table 11(a,b). However, the `Orientation` may be known in two cases:

1. When the noun contains an exponent which shows that it results from a derivational process that cannot yield a verb, then the noun is the base of the conversion and the verb derives from it. For instance, in Table 11(c), the noun is a neoclassical compound, and in French, neoclassical compounding never produces verbs. Therefore, *hydrogéner*$_{\mathrm{V}}$ derives from *hydrogène*$_{\mathrm{Nmas}}$.
2. Symmetrically, when the verb contains a formal mark showing that it results from an affixation, and that this affixation cannot yield a noun, then the verb is the base. In Table 11(d), the intensive prefixation in *dé-* only produces verbs. Therefore, the noun *découpe* derives form the verb *découper*.

We can use the exact same features to describe cross-formation (Table 1(d)). The value `indirect` of `Orientation` indicates that both lexemes have exponents and that they are substituted one for the other. Table 12(a, b, c) describes the relations between the members of the (*race*, *racisme*, *raciste*) family, and Table 12(d) the (*fascisme*, *fasciste*) relation. In Table 12(c, d), the values of `Orientation`, `Complexity`, and `Morpho(logical) Match(ing)` are the same for the cross-formation relations (*fascisme, fasciste*) and (*racisme, raciste*). This shows that the incomplete family of (*fascisme, fasciste*) belongs to a sub-paradigm of the paradigm of (*race, raciste, racisme*).

Back-formation can also very easily be described by means of Démonette's features as in Table 13(a, b, c). The verbs (L_1) and the nouns (L_2) start with the same neoclassical components (*thermo-* 'heat', *hydro-* 'water' and *aéro-* 'air'). In addition, the nouns are suffixed by *-age*. Formally, the nouns are more complex than the corresponding verb as indicated by the value `X/Xage` for `Morph Match` which also describes regular suffixation in *-age* as in Table 13(d). However, for the back-formations of Table 13(a, b, c), the value of `Orientation` is `d2a` and not `a2d` as in regular derivations (Table 13(d)). This value expresses the fact that the verb is derived from the noun, for the same reason as with *hydrogène*$_{\mathrm{Nmas}}$→*hydrogéner*$_{\mathrm{V}}$ in Table 11(c): the nouns in Table 13(a, b, c) are formed by neoclassical compounding, like *collage*→*thermocollage* (Table 13(e)) and neoclassical compounding cannot not yield verbs in French. This means that the verbs in Table 13(a, b, c) are derived from the (formally more complex) nouns.

	L_1	L_2	Orientation	Complexity
a	*scier*$_{\mathrm{V}}$ 'to saw'	*scie*$_{\mathrm{Nfem}}$ 'saw'	nd	simple
b	*danser*$_{\mathrm{V}}$ 'to dance'	*danse*$_{\mathrm{Nfem}}$ 'dance'	nd	simple
c	*hydrogéner*$_{\mathrm{V}}$ 'to hydrogenate'	*hydrogène*$_{\mathrm{Nmas}}$ 'hydrogene'	d2a	simple
d	*découper*$_{\mathrm{V}}$ 'to cut out'	*découpe*$_{\mathrm{Nfem}}$ 'cut'	a2d	simple

Table 11. Démonette. Verb-noun conversion

	L_1	L_2	Schema_{L1}	Schema_{L2}	Morph Match	Orientation	Complexity
a.	*race*	*racisme*	X	Xisme	X/Xisme	a2d	simple
b.	*race*	*raciste*	X	Xiste	X/Xiste	a2d	simple
c.	*racisme*	*raciste*	Xisme	Xiste	Xisme/Xiste	indirect	simple
d.	*fascisme*	*fasciste*	Xisme	Xiste	Xisme/Xiste	indirect	simple

Table 12. Démonette. Cross formation: Structural and formal properties of race/raciste/racisme *and* fascisme/fasciste *families*

4.2.2. Semantic features

Démonette provides a semantic description for the relations where Complexity =simple. It includes the semantic type of the relation (SemRel), a gloss in natural language which defines L_1 and L_2 with respect to each other (Concrete Definition) and a generalization of this cross-definition (Abstract Definition). Table 14 presents examples of these semantic description.

The value of SemRel depends on a combination of features that describe L_1, L_2 and their relation: the ontological classes of L_1 and L_2; the parts-of-speech of L_1 and L_2; the Orientation, Complexity and Morph(ological) Match(ing) of the relation. Different combinations may correspond to the same value of SemRel.

The values for SemRel in Table 14 are syn(onymy), iter(ativity) and pot(entiality). The value is syn(onymy) for relations between a dynamic predicate and its derived action noun (when both denote dynamic situations (Onto.Type=Dyn-Situation) and when the morphological properties of the relation describe a direct base-derivative derivation rule, e.g. Morph Match=X/Xage as in Table 14(a, b)). SemRel equals iter (ativity) when the value of Morph Match involves an iterative prefixation like *re-* (reX/X in Table 14(c, d) or reX/Xage in Table 14(e)). Its value is pot(entiality) when L_1 or L_2 denotes a dynamic predicate and the other lexeme denotes an *-able* suffixed modifier, as in Table 14(f, g, h).

	L_1	L_2	Morph Match	Orientation	Complexity
a.	*thermocoller*	*thermocollage*	X/Xage	d2a	simple
b.	*hydromasser*	*hydromassage*	X/Xage	d2a	simple
c.	*aérosonder*	*aérosondage*	X/Xage	d2a	simple
d.	*coller*	*collage*	X/Xage	a2d	simple
e.	*thermocollage*	*collage*	thermoX/X	d2a	simple

Table 13. Démonette. Representation of Back Formation

	L$_1$ & Ont.Type$_{L1}$	L$_2$ & Ont.Type$_{L2}$	Morph Match	Sem Rel	Concrete Definition & Abstract Definition
a.	*laver*$_{V1}$ Dyn-Sit$_{V1}$	*lavage*$_{N2}$ Dyn-Sit$_{N2}$	X/Xage	syn	'laver$_{V1}$ sth is to perform lavage$_{N2}$ on it' 'Dyn-Sit$_{V1}$ sth is to perform Dyn-Sit$_{N2}$'
b.	*relaver*$_{V1}$ Dyn-Sit$_{V1}$	*relavage*$_{N2}$ Dyn-Sit$_{N2}$	X/Xage	syn	'relaver$_{V1}$ sth is to perform relavage$_{N2}$ on it' 'Dyn-Sit$_{V1}$ sth is to perform Dyn-Sit$_{N2}$'
c.	*laver*$_{V1}$ Dyn-Sit$_{V1}$	*relaver*$_{V2}$ Dyn-Sit$_{V2}$	X/reX	iter	'laver$_{V1}$ sth several times is to relaver$_{V2}$ it' 'Dyn-Sit$_{V1}$ sth several times is to Dyn-Sit$_{V2}$ it'
d.	*lavage*$_{N1}$ Dyn-Sit$_{N1}$	*relavage*$_{N2}$ Dyn-Sit$_{N2}$	X/reX	iter	'Perform several lavage$_{N1}$ is to perform relavage$_{N2}$' 'Perform several Dyn-Sit$_{N1}$ is to perform Dyn-Sit$_{N2}$'
e.	*lavage*$_{N1}$ Dyn-Sit$_{N1}$	*relaver*$_{V2}$ Dyn-Sit$_{V2}$	Xage/reX	iter	'Perform several lavage$_{N1}$ is to relaver$_{V2}$' 'Perform several Dyn-Sit$_{N1}$ is to Dyn-Sit$_{V2}$'
f.	*laver*$_{V1}$ Dyn-Sit$_{V1}$	*lavable*$_{A2}$ Mod$_{A2}$	X/Xable	pot	'One can laver$_{V1}$ sth if it is lavable$_{A2}$' 'One can Dyn-Sit$_{V1}$ sth if it is Mod$_{A2}$'
g.	*lavage*$_{N1}$ Act$_{N1}$	*lavable*$_{A2}$ Mod$_{A2}$	Xage/Xable	pot	'One can perform lavage$_{N1}$ on sth if it is lavable$_{A2}$' 'One can perform Dyn-Sit$_{N1}$ on sth if it is Mod$_{A2}$'
h.	*relaver*$_{V1}$ Dyn-Sit$_{V1}$	*lavable*$_{A2}$ Mod$_{A2}$	reX/Xable	pot	'One can relaver$_{V1}$ sth if it is lavable$_{A2}$ several times' 'One can Dyn-Sit$_{V1}$ sth if it is Mod$_{A2}$ several times'

Table 14. Démonette. Semantic features of the relations in the family of laver

The values of Concrete Definition are inspired by Frame Semantics tradition and especially *FrameNet*, its most popular implementation (Fillmore, 2006). The fundamental assumption is that people understand language through situations evoked in their mind by certain words. These representations are called frames, and involve the participants to the situation. Unlike frames, the situations described in the Concrete Definition glosses are derivationally relevant but may not be relevant cognitively; see (Sanacore et al., 2019).

The Abstract Definitions are generalizations of the Concrete Definitions where L$_1$ and L$_2$ are replaced by their ontological types. For instance, in Table 14(g), the Concrete Definition of (*lavage*$_{N1}$, *lavable*$_{A2}$) is 'One can perform *lavage*$_{N1}$ on something if it is *lavable*$_{A2}$'; in the corresponding Abstract Definition, *lavage*$_{N1}$ replaced by Dyn-Sit$_{N1}$ and *lavable*$_{A2}$ by Mod$_{A2}$. Derivational relations with the same Abstract Definition belong to the same semantic series like (*laver*, *lavage*) in Table 14(a) and (*relavage*, *relaver*) in Table 14(b)[5].

Table 15 presents an example of the description of rival WF processes in Démonette. The derivational relations listed in this Table are the same as in Table 9 (columns

[5] The semantic features of symmetrical pairs (L$_1$, L$_2$) and (L$_2$, L$_1$) are identical when their indexes are switched.

1 and 2). They involve competing WF processes because their descriptions are identical except for `Schema`$_{L1}$ or `Schema`$_{L2}$ (and consequently for `Morphological Matching`) and of course for `Concrete Definition` because the lexemes are different. The identical features are omitted: `Orientation=a2d`, `Complexity=simple`, `Onto.Type=Dyn-Sit`, `SemRel=syn(onymy)`, and the value of `Abstract Definition`, i.e. `` `Dyn-Sit``$_{V1}$ `sth is to perform Dyn-Sit`$_{N2}$`'`.

L$_1$	L$_2$	Schema$_{L1}$	Schema$_{L2}$	Concrete Definition
laver	*lavage*	X	Xage	‘`laver`$_{V1}$ sth is to perform `lavage`$_{N2}$’
classer	*classement*	X	Xment	‘`classer`$_{V1}$ sth is to perform `classement`$_{N2}$’
planter	*plantation*	X	Xation	‘`planter`$_{V1}$ sth is to perform `plantation`$_{N2}$’
souder	*soudure*	X	Xure	‘`souder`$_{V1}$ sth is to perform `soudure`$_{N2}$’
découper	*découpe*	X	X	‘`découper`$_{V1}$ sth is to perform `découpe`$_{N2}$’

Table 15. Démonette. Affix rivalry

4.3. Meaning-form discrepancies in **Démonette**

We saw how Démonette’s set of features can be used to represent almost any type of derivation: regular affixation (*laver→lavage*), conversion, back-formation and affix rivalry. The independence between semantic descriptions (e.g. `SemRel`), morphological structures (e.g. `Morphological Matching`) and structural properties of relations (e.g. `Orientation`) is the key to the descriptive power of this set of features.

With these features, it is also possible to describe the asymmetrical parasynthetic constructions presented in Table 1(b) and in Table 6. The description of these meaning-form discrepancies only requires the addition of two values, `formal-motivation` and `semantic-motivation`, to the feature `Complexity`. Table 16 shows how these values are used[6].

As we discussed above (§ 3.4), parasynthetic formations have distinct formal and semantic motivations. For instance, the forms of *multicellulaire* and *pluricellulaire* are derived from the form of *cellulaire* and their meaning is derived from the meaning of *cellule*. For these parasynthetic forms, the description in Démonette is split into two entries, one for the formal motivation (Table 16(d, e)) and the other for the semantic motivation (Table 16(b, c)). In the first, `Complexity` has the `form(al)-motiv(ation)` value and the semantic features are all left blank. In the other, the semantic relation is `plurality` and the value of `Complexity` is `sem(antic)-motiv(ation)`. This value indicates that the relation is semantically grounded but it is not morphologi-

[6] For the sake of space, the relations are listed in only one direction.

	L_1	L_2	Morph Match	Orientation	Complexity	SemRel
a.	*cellule*	*cellulaire*	X/Xaire	a2d	simple	relation
b.	*cellule*	*multicellulaire*	X/multiXaire	a2d	sem-motiv	plurality
c.	*cellule*	*pluricellulaire*	X/pluriXaire	a2d	sem-motiv	plurality
d.	*cellulaire*	*multicellulaire*	X/multiX	a2d	form-motiv	—
e.	*cellulaire*	*pluricellulaire*	X/pluriX	a2d	form-motiv	—
f.	*pluricellulaire*	*multicellulaire*	pluriX/multiX	indirect	simple	synonymy

Table 16. Démonette. Description of the parasynthetic relations in the family of cellule$_N$

cally: the values of the feature Morpho(logical) Match(ing) (i.e. X/multiXaire and X/pluriXaire) are not used for regular derivations.

With the values sem-motiv and form-motiv, Démonette can independently represent relations in the formal and semantic paradigms just as in ParaDis and thus becomes a large-scale formalization of this model: a relation with Complexity=form-motiv only belongs to the formal network (no semantic counterpart) while a relation with Complexity=sem-motiv only belongs to the semantic network. The other values for Complexity, that is, simple and complex, characterize compositional relations: for instance, the base/derivative regular relation (*cellule*, *cellulaire*) in Table 16(a), and the indirect, prefix replacement relation in the pair of synonyms (*multicellulaire*, *pluricellulaire*) in Table 16(f).

5. Conclusion

In this article, we have presented Démonette and its theoretical background. The resource is under development and many of the results we discussed are still partial. Our goal is to provide a semantically and formally homogeneous description of French derivational morphology, both regular and non-canonical, by combining principles taken from lexeme-based morphology and paradigmatic models of derivation.

We have shown throughout this article that Démonette and ParaDis are actually two sides of the same project. One of the benefits of their joint development is a decisive and mutual enrichment of the two sides. They largely have the same goal which is to model and describe French derivational paradigms. Ultimately, Démonette will provide a playground where all sorts of hypotheses may be tested. Another goal is to provide an effective answer to the question "what does a derivational paradigm look like?". On the other hand, ParaDis addresses the same question from a different angle: "How does paradigmatic derivational morphology work and why do we need it?". The success of this effort owes much to Démonette which helped clarify many ideas morphologists had about derivational paradigms and identify the main principles articulated in ParaDis.

Démonette has a simple, robust and highly redundant representation format where many existing morphological descriptions can be reframed. It is purely relational and only describes the WF processes through the pairs of lexemes they help form. One consequence of the parallel development of Démonette and ParaDis is the importance given to the non-canonical formations in the two sides of the project. Actually, most of the progress brought by this effort comes from the need to have a clean description of the analysis of these formations. It also results in an imbalance in ParaDis where the representational component is fully fledged while the processive one (i.e. the inventory of the constraints that control the filling of the paradigms) remains sketched. Démonette and ParaDis have very similar scopes in terms of phenomena and morphological processes, with one exception: composition. Composition cannot be described in ParaDis because it does not fit in the derivational paradigms defined by the affixations, conversions and all their non-canonical variants. On the other hand, the relations between a compound and its components can easily be represented in Démonette (by means of an additional value `composition` of the feature `Complexity`).

Acknowledgements

This work benefited from the support of the project DEMONEXT ANR-17-CE23-0005 of the French National Research Agency (ANR). We thank the participants in the DEMONEXT project for their contribution to the results presented in this paper.

Bibliography

Amiot, Dany. Plusieurs versus poly-, pluri- et multi-. *Verbum*, 27(4):403–417, 2005.

Anderson, Stephen R. *A-Morphous Morphology*. Cambridge University Press, Cambridge, UK, 1992. doi: 10.1017/CBO9780511586262.

Antoniova, Vesna and Pavol Štekauer. Derivational paradigms within selected conceptual fields – contrastive research. *Facta Universitatis, Series: Linguistics and Literature*, 13(2):61–75, 2015.

Aronoff, Mark. *Word Formation in Generative Grammar*. Linguistic Inquiry Monographs. MIT Press, Cambridge, MA, 1976.

Baayen, R. Harald, Richard Piepenbrock, and Leon Gulikers. The CELEX Lexical Database (Release 2). CD-ROM, 1995. URL `https://catalog.ldc.upenn.edu/LDC96L14`. Linguistic Data Consortium, Philadelphia, PA.

Becker, Thomas. Back-formation, cross-formation, and 'bracketing paradoxes' in paradigmatic morphology. In Booij, Geert E. and Jaap van Marle, editors, *Yearbook of Morphology 1993*, pages 1–25. Kluwer Academic Publishers, Dordrecht, 1994. doi: 10.1007/978-94-017-3712-8_1.

Bernhard, Delphine, Bruno Cartoni, and Delphine Tribout. A Task-Based Evaluation of French Morphological Resources and Tools. *Linguistic Issues in Language Technology*, 5(2), 2011. URL `https://halshs.archives-ouvertes.fr/halshs-00746391`.

Blevins, James P. *Word and paradigm morphology*. Oxford University Press, 2016. doi: 10.1093/acprof:oso/9780199593545.001.0001.

Bochner, Harry. *Simplicity in generative morphology*. Mouton de Gruyter, Berlin & New-York, 1993. doi: 10.1515/9783110889307.

Bonami, Olivier and Jana Strnadová. Paradigm structure and predictability in derivational morphology. *Morphology*, 29(2):167–197, 2019. doi: 10.1007/s11525-018-9322-6.

Booij, Geert. *Construction Morphology*. Oxford University Press, Oxford, 2010. doi: 10.1093/acrefore/9780199384655.013.254.

Booij, Geert and Francesca Masini. The role of second order schemas in the construction of complex words. In Bauer, Laurie, Lívia Körtvélyessy, and Pavol Štekauer, editors, *Semantics of complex words*, volume 47, pages 47–66. Springer, Heidelberg, 2015. doi: 10.1007/978-3-319-14102-2_4.

Corbett, Greville G. Canonical derivational morphology. *Word Structure*, 3(2):141–155, 2010. doi: 10.3366/word.2010.0002.

Cotterell, Ryan and Hinrich Schütze. Joint Semantic Synthesis and Morphological Analysis of the Derived Word. *CoRR*, abs/1701.00946, 2017. doi: 10.1162/tacl_a_00003. URL `http://arxiv.org/abs/1701.00946`.

Creutz, Mathias and Krista Lagus. Unsupervised Morpheme Segmentation and Morphology Induction from Text Corpora Using Morfessor 1.0. Technical Report A81, Helsinki University of Technology, 2005.

Dahl, Eystein and Antonio Fábregas. Zero Morphemes. In *Oxford Research Encyclopedia of Linguistics*. Oxford University Press, 2018. doi: 10.1093/acrefore/9780199384655.013.592.

Fellbaum, Christiane, Anne Osherson, and Peter E. Clark. Putting semantics into WordNet's "morphosemantic" links. In *Human Language Technology. Challenges of the Information Society*, volume 5603 of *Lecture Notes in Computer Science Volume*, pages 350–358. Springer, 2009. doi: 10.1007/978-3-642-04235-5_30.

Fillmore, Charles. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences*, 280:20 – 32, 12 2006. doi: 10.1111/j.1749-6632.1976.tb25467.x.

Fradin, Bernard. *Nouvelles approches en morphologie*. PUF, Paris, 2003. doi: 10.3917/puf.fradi.2003.01.

Goldsmith, John. Unsupervised Learning of the Morphology of Natural Language. *Computational Linguistics*, 27(2):153–198, 2001. doi: 10.1162/089120101750300490.

Habash, Nizar and Bonnie Dorr. A categorial variation database for English. In *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference (NAACL/HLT 2003)*, pages 96–102, Edmonton, 2003. ACL. doi: 10.3115/1073445.1073458.

Hathout, Nabil. Acquisition of morphological families and derivational series from a machine readable dictionary. In Montermini, Fabio, Gilles Boyé, and Jesse Tseng, editors, *Selected Proceedings of the 6th Décembrettes: Morphology in Bordeaux*, Somerville, MA, 2009. Cascadilla Proceedings Project. URL `https://arxiv.org/abs/0905.1609`.

Hathout, Nabil. Une approche topologique de la construction des mots : propositions théoriques et application à la préfixation en *anti-*. In *Des unités morphologiques au lexique*, pages 251–318. Hermès Science-Lavoisier, Paris, 2011.

Hathout, Nabil and Fiammetta Namer. Discrepancy between form and meaning in Word Formation: the case of over- and under-marking in French. In Rainer, Franz, Wolfgang U. Dressler, Francesco Gardani, and Hans Christian Luschützky, editors, *Morphology and meaning*, pages 177–190. John Benjamins, Amsterdam, 2014a. doi: 10.1075/cilt.327.12hat.

Hathout, Nabil and Fiammetta Namer. Démonette, a French derivational morpho-semantic network. *Linguistic Issues in Language Technology*, 11(5):125–168, 2014b.

Hathout, Nabil and Fiammetta Namer. Giving Lexical Resources a Second Life: Démonette, a Multi-sourced Morpho-semantic Network for French. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, 2016. URL http://www.lrec-conf.org/proceedings/lrec2016/pdf/279_Paper.pdf.

Hathout, Nabil and Fiammetta Namer. La parasynthèse à travers les modèles : des RCL au ParaDis. In Bonami, Olivier, Gilles Boyé, Georgette Dal, Hélène Giraudo, and Fiammetta Namer, editors, *The lexeme in descriptive and theroretical morphology*, Empirically Oriented Theoretical Morphology and Syntax, pages 365–399. Language science Press, Berlin, 2018. URL http://langsci-press.org/catalog/book/165.

Hockett, Charles Francis. Two models of grammatical description. *Words*, 10:210–234, 1954. doi: 10.1080/00437956.1954.11659524.

Iacobini, Claudio. Parasynthesis in Morphology. In *Oxford Research Encyclopedia of Linguistics*. Oxford University Press, Oxford, 2020. doi: 10.1093/acrefore/9780199384655.013.509.

Koenig, Jean-Pierre. *Lexical Relations*. CSLI Publications, Stanford, CA, 1999.

Kyjánek, Lukáš. Morphological Resources of Derivational Word-Formation Relations. Technical Report 61, ÚFAL - Charles University, Prague, 2018.

Lafourcade, Mathieu and Alain Joubert. JeuxDeMots : un prototype ludique pour l'émergence de relations entre termes. In *JADT'08 : Journées internationales d'Analyse statistiques des Données Textuelles*, pages 657–666, France, March 2008. URL https://hal-lirmm.ccsd.cnrs.fr/lirmm-00358848.

Litta, Eleonora, Marco Passarotti, and Chris Culy. Formatio formosa est. Building a Word Formation Based Lexicon for Latin. In *Proceedings of the Third Italian Conference on Computational Linguistics, CLiC-it 2016*, pages 185–189, Naples, Italy, 2016. URL http://ceur-ws.org/Vol-1749/paper32.pdf.

Manova, Stela. Subtraction. In Manova, Stela, editor, *Understanding Morphological Rules - With Special Emphasis on Conversion and Subtraction in Bulgarian, Russian and Serbo-Croatian*, volume 1, pages 125–172. Springer, Dordrecht, 2011. doi: https://doi.org/10.1007/978-90-481-9547-3.

McCarthy, John and Alan Prince. Prosodic Morphology: Constraint Interaction and Satisfaction. Technical report 3, Rutgers University Center for Cognitive Science, New Brunswick, NJ, 1993. URL https://scholarworks.umass.edu/cgi/viewcontent.cgi?article=1013&context=linguist_faculty_pubs.

Mel'čuk, Igor. Lexical Functions: A Tool for the Description of Lexical Relations in the Lexicon. In Wanner, Leo, editor, *Lexical Functions in Lexicography and Natural Language Processing*, pages 37–102, Amsterdam/Philadelphia, 1996. John Benjamins.

Miller, Georges A., Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4):335–391, 1990. doi: 10.1093/ijl/3.4.235.

Namer, Fiammetta. *Morphologie, lexique et traitement automatique des langues : L'analyseur DériF*. Hermès Science-Lavoisier, Paris, 2009.

Namer, Fiammetta. A Rule-Based Morphosemantic Analyzer for French for a Fine-Grained Semantic Annotation of Texts. In Mahlow, Cerstin and Michael Piotrowski, editors, *SFCM 2013*, CCIS 380, pages 93–115. Springer, Heidelberg, 2013. doi: 10.1007/978-3-642-40486-3_6.

Namer, Fiammetta, Nabil Hathout, and Stéphanie Lignon. Adding morpho-phonology into a French morpho-semantic resource: Demonette. In Litta, Eleonora and Marco Passarotti, editors, *Proceedings of the First Workshop in Resources and Tools for Derivational Morphology (DeriMo),*, pages 49–60, Milano, Italy, oct 2017. EDUCatt. ISBN 978-88-9335-225-3.

Sanacore, Daniele, Nabil Hathout, and Fiammetta Namer. Semantic descriptions of French derivational relations in a families-and-paradigms framework. In Žabokrtský, Zdeněk, Magda Ševčíková, Eleonora Litta, and Marco Passarotti, editors, *Derimo - Second International Workshop on Resources and Tools for Derivational Morphology*, pages 15–24. Charles University, Prague, 2019. ISBN ISBN 978-80-88132-08-0.

Ševčíková, Magda, Adéla Kalužová, and Zdeněk Žabokrtský. Identification of aspectual pairs of verbs derived by suffixation in the lexical database DeriNet. In *Proceedings of the Workshop on Resources and Tools for Derivational Morphology (DeriMo)*, pages 105–116, Milano, Italy, 2017. EDUCatt. ISBN 978-88-9335-225-3.

Shafaei, Elnaz, Diego Frassinelli, Gabriella Lapesa, and Sebastian Padó. DErivCELEX: Development and Evaluation of a German Derivational Morphology Lexicon based on CELEX. In Litta, Eleonora and Marco Passarotti, editors, *Proceedings of the First Workshop in Resources and Tools for Derivational Morphology (DeriMo),*, pages 117–128, Milano, Italy, oct 2017. EDUCatt. ISBN 978-88-9335-225-3.

Šojat, Krešimir, Matea Srebačić, Marko Tadić, and Tin Pavelić. CroDeriV: a new resource for processing Croatian morphology. In Chair), Nicoletta Calzolari (Conference, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may 2014. European Language Resources Association (ELRA). ISBN 978-2-9517408-8-4.

Spencer, Andrew. *Lexical relatedness*. Oxford University Press, Oxford, 2013. doi: 10.1093/acprof:oso/9780199679928.003.0003.

Steiner, Petra and Josef Ruppenhofer. Building a Morphological Treebank for German from a Linguistic Database. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 3882–3889, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL `https://www.aclweb.org/anthology/L18-1613`.

Štekauer, Pavol. Derivational Paradigms. In Lieber, Rochelle and Pavol Štekauer, editors, *The Oxford Handbook of Derivational Morphology*, pages 354–369. Oxford, Oxford University Press, Oxford, 2014. doi: 10.1093/oxfordhb/9780199641642.013.0020.

Talamo, Luigi, Chiara Celata, and Pier Marco Bertinetto. DerIvaTario: An annotated lexicon of Italian derivatives. *Word Structure*, 9(1):72–102, 2016. doi: 10.3366/word.2016.0087.

Thornton, Anna M. Reduction and maintenance of overabundance. A case study on Italian verb paradigms. *Word Structure*, 5(2):183–207, 2012. doi: 10.3366/word.2012.0026.

Todaro, Giuseppina. *Nomi (e aggettivi) che diventano verbi tramite prefissazione: quel che resta della parasintesi*. PhD thesis, Tesi di dottorato, Università Roma Tre et Université Toulouse Jean-Jaurès, 2017.

Tribout, Delphine. Verbal stem space and verb to noun conversion in French. *Word Structure*, 5 (1):109–128, 2012. doi: 10.3366/word.2012.0022.

Vidra, Jonáš, Zdeněk Žabokrtský, Lukáš Kyjánek, Magda Ševčíková, and Šárka Dohnalová. DeriNet 2.0, 2019. URL `http://hdl.handle.net/11234/1-2995`. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Žabokrtský, Zdeněk, Magda Ševčíková, Milan Straka, Jonáš Vidra, and Adéla Limburská. Merging Data Resources for Inflectional and Derivational Morphology in Czech. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1307–1314, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL `https://www.aclweb.org/anthology/L16-1208`.

Zeller, Britta D, Jan Snajder, and Sebastian Padó. DErivBase: Inducing and Evaluating a Derivational Morphology Resource for German. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1201–1211, Sofia, Bulgaria, 2013. URL `https://www.aclweb.org/anthology/P13-1118.pdf`.

Address for correspondence:
Fiammetta Namer
`fiammetta.namer@univ-lorraine.fr`
Université de Lorraine, CLSH
UFR SHS, dépt SDL
23 bd Albert 1er, BP 60446
54001 NANCY CEDEX, France

The Prague Bulletin of Mathematical Linguistics
NUMBER 114 APRIL 2020 35–57

Using Word Embeddings and Collocations for Modelling Word Associations

Micha de Rijk, David Mareček

Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics, Charles University

Abstract

Word association is an important part of human language. Many techniques for capturing semantic relations between words exist, but their ability to model word associations is rarely tested in a real application. In this paper, we evaluate three models aimed at different types of word associations: a word-embedding model for synonymy, a point-wise mutual information model for word collocations, and a dependency model for common properties of words. The quality of the proposed models is tested on English and Czech by humans in an online version of the word-association game "Codenames".

1. Introduction

Association is one of the basic mechanisms of human memory. It is based on the past experience of a man and existing relationship between the phenomena of the real world (Morkovkin, 1970). A well-known word game called "Word associations" involves an exchange of words that are associated together. Its idea is based on the connection and production of other words in spontaneous response to a given word. In another version of the game, the associations between words must be strictly obvious, rather than the usual "first word that comes to mind", which can often require explaining how it is connected with the previous word.

Word associations can also be used in psychology or psychiatric evaluations. Jung (1910) theorized that people connect ideas, feelings, experiences and information by way of associations. Gough (1976) believes that word association can reveal something of a person's subconscious mind as it shows what things they associate together.

Corresponding author: marecek@ufal.mff.cuni.cz

Cite as: Micha de Rijk, David Mareček. Using Word Embeddings and Collocations for Modelling Word Associations. The Prague Bulletin of Mathematical Linguistics No. 114, 2020, pp. 35–57. doi: 10.14712/00326585.002.

Words can be associated in many different ways, some of them are listed in the following overview:

- **Synonyms**: A synonym is a word that expresses the same concept as another word. For example, *drink* is a synonym of*beverage*.
- **Hypernyms**: We say that A is a hypernym of B if A describes a set of concepts that B belongs to. For example, *fruit* is a hypernym of *apple* because *apple* belongs to the set of objects described by the word *fruit*.
- **Hyponyms**: The opposite of a hypernym. The word *apple* is a hyponym of *fruit* because *apple* is a type of *fruit*.
- **Co-hyponyms**: The co-hyponym relation refers to words that have a hypernym in common such as *knife* and *fork* which are both hyponyms of the word *utensil*.
- **Meronyms**: A meronym is a word that is a part or member of another. For example, *sentence* is a meronym of *text* because a sentence is usually part of a text.
- **Holonyms**: A holonym is the opposite of a meronym. *Text* is a holonym of the word *sentence*.
- **Collocate**: A collocation is a set of words that co-occur more often than would be expected by random chance. The individual members of such a set are called collocates. For example, the individual words *code* and *source* are collocates because of their frequent co-occurrence in the compound word *source code*.

Word associations can also vary in strength based on the direction of the association. For example, the word *Eiffel* would be very strongly related to the word *tower*: when someone says *Eiffel*, *tower* immediately springs to mind. However, this relation does not hold as strongly when inverted. If someone says *tower*, words like *building* and *tall* spring to mind much more quickly than *Eiffel*. Similarly, *brick* is related to *tower*, but not to *Eiffel*. As such, word association cannot be treated as a symmetric or transitive relation.

Modelling these different types of word associations computationally is very challenging. There are many ways in which words can be associated. Gathering all of these associations for each individual word in a language is an immense task. In fact, we argue that it is infeasible to encode all such relations in manually constructed ontologies and databases. Two of them are given in Section 2.

Instead, in Section 3, we propose three unsupervised methods for modelling different types of word associations using large amounts of text as a source: a word-embedding model for synonymy, a point-wise mutual information model for word collocations, and a dependency model for common properties of words.

The main goal of this work is to evaluate the performance of the proposed methods. Since we do not have any appropriate annotated data, we test our models directly by humans through a simplified online game called "Codenames". It is a single-player version of a very popular word association board game of the same name. In short, one player gives one-word hints to some of the words given on the board and the other player guesses. The game itself and the evaluation procedure is described in Section 4.

In Section 5 we describe methods on how to employ the association measures in the Codenames game. In Section 6 we provide a detailed analysis of the performance of our models and two ensemble models which try to combine all the models together. A concise overview of our findings is given in Section 7. We also mention several promising directions for future research.

2. Association Databases

One of the approaches for computational word association that we considered is the use of ontologies and databases. We could rely completely on the word associations provided by manually entered data to build our models. Even though we finally decided not to use them in our system, we detail two of these approaches below.

2.1. WordNet

WordNet (Fellbaum, 1998) is a collection of synsets grouped into a semantic hierarchy. A synset is a collection of one or more words with the same meaning, i.e. synonyms. Because of the information it encodes, it excels at strict relations such as hypernyms, hyponyms, meronyms and holonyms. This would be a great addition to our application and a fruitful area for future research on computational models of word associations. For example, *continent* would be a useful hint for both *Africa* and *Australia*. However, it falls short when considering more free associations such as *Frodo* and *ring*, which cannot be classified as either hypernym, hyponym, meronym or holonym and is thus not captured in WordNet.

WordNet is not suitable for our purposes because it does not capture as many relations as we would like and is not as extensible as data-driven methods, which are able to capture even pop culture references such as the relation between *Frodo* and *ring*.

2.2. University of South Florida Free Association Norms

The University of South Florida Free Association Norms (Nelson et al., 2004) is a database of free associations containing 72,000 word pairs collected from almost 750,000 responses produced by over 6,000 participants. More than 5,000 stimulus words were tested. While this is a great resource for human responses on word association, it has too many gaps to be suitable for a computational model. Even when we look at all the responses in addition to the 5,000 stimulus words, words that occur in the original Codenames board game such as *Amazon*, *Greece* and *horseshoe* do not occur in the database at all. These gaps can only be filled by repeating the experiment with these words as stimulus words. Moreover, this database exists only for English, limiting the applicability of this approach for other languages.

Although it is not suitable as a basis for a complete computational model, it is still useful as a resource on human word association. The database of word associations could be used to compare how similar the predictions made by a computational model are to human-level associations. While we do not perform this particular comparison ourselves, it could serve as an interesting evaluation metric for future work.

We think ontologies and association databases contain too many blind spots and often fail to encode unorthodox or out-of-the-box relations that would nonetheless be considered valid associations by humans. Building these resources also requires a lot of manual work and knowledge of the language involved, which becomes a recurring cost as semantic shifts occur in the existing vocabulary and new words enter the language. As such, we turn our focus towards automated methods for the extraction of word associations in the rest of this paper.

2.3. Other Related Works

Thawani et al. (2019) propose a novel word embeddings evaluation task by employing a large word association dataset called *Small World of Words* (De Deyne et al., 2018). It contains Word association and participant data for 100 primary, secondary, and tertiary responses to 12,292 cues, collected from over 90,000 participants.[1]

3. Methods

In this section, we propose three data-driven methods that can be used for measuring how much two words may be associated each to other.

We are not aware of any work in which more complex word-association models were built. We know of only one earlier attempt in this area, which is a Master thesis by Obrtlík (2018). However, they use word embeddings, which cover only synonymic relations. We describe this method in Section 3.1. Human word associations are not limited to this type of relation alone. Take, for example, the words *ice* and *cream* in a collocate such as *ice cream*. Therefore, in Sections 3.2 and 3.3, we propose two other methods based on word collocations.

3.1. Word Embeddings

Word embeddings are vector representations of words that are used in neural networks processing of textual data. They capture semantic similarity: words that occur in a similar context have a similar meaning and are grouped together in the word-embeddings vector space. Word embeddings capture synonymy, which makes them useful for word association. To create such embeddings efficiently, Mikolov et al. (2013) introduce the skip-gram model with negative sampling. Since then many additions to this technique have been proposed, such as adding topic information (Liu

[1] `https://smallworldofwords.org/en/project`

et al., 2015) and deriving the embeddings from dependency-based contexts (Levy and Goldberg, 2014).

For our word-association model, we use word embeddings enriched with subword information as described by Bojanowski et al. (2017). This method is called *fastText* and adapts the skip-gram model with negative sampling to represent a word as a combination of the character n-grams it contains. The benefit of this approach is that the representations of character n-grams are global and shared between all words, so more accurate representations for rare words are obtained. The *fastText* embeddings are available in many languages, which makes it easier to build the same model for other languages.[2]

The pre-trained model provides over 2.5 million word embeddings for English and 600,000 for Czech. We cut down on the size of this collection considerably to limit the computation time needed to compare against all of these embeddings when scoring a word. The model is ordered according to the word frequencies. To avoid clutter and make sure that we do not include words that people might not know, we limit the number of word embeddings in our model to the top $10,000$ words, sorted by their unigram frequency in Wikipedia.

For measuring similarity of two given words a and b represented by word embeddings A and B we compute the cosine distance as follows:

$$\text{cosine\_distance}(a, b) = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

3.2. Sentence-level Collocations

A different type of word association may be covered by computing word co-occurrences. We can see this when we look at the collocation *Eiffel Tower*. When somebody says *"Eiffel"*, we quickly think *"Tower"*.

To find these collocations we need a large amount of text and a measure of association. The text is taken from the training sections of the CzEng 1.7 corpus (Bojar et al., 2016).[3] CzEng is a large parallel corpus for Czech and English, containing roughly 57 million sentence pairs and over 600 million words. The corpus bundles a large amount of data, including but not limited to text from subtitles, EU legislation, fiction and web pages.

For measuring word collocations we use pointwise mutual information (PMI)

$$\text{PMI}(a, b) = \log_2 \frac{p(a, b)}{p(a)p(b)},$$

[2] https://github.com/facebookresearch/fastText/blob/master/docs/pretrained-vectors.md

[3] http://ufal.mff.cuni.cz/czeng/

where $p(a,b)$ is the probability that *b* occurs after *a* and $p(a)$ is the probability of seeing *a*. For practical purposes we make a slight modification to the usual definition of PMI and say that $p(a,b)$ is the number of times *b* occurs **before or after** a. This way the direction of the relation does not matter for the strength of the association.[4]

Experimentally we find that simple bigrams as described above do not form a good model. Finding a good collocate for one word is easily done, but finding good collocates for multiple words, is rarely successful because there is too little overlap between collocates. What is a frequent collocate for one word, is almost never a frequent collocate for another. The problem here is that related words are often separated by function words in the text, which means they do not form a bigram and are not seen as a collocation.

To solve this we propose a collocation model over sentence-level word pairs. We define $p(a,b)$ as the probability that *a* and *b* occur in the same sentence. This provides a much broader scope for co-occurrences, which increases the chance of overlapping collocates when trying to find a high scoring collocate for multiple words. The upside of sentence-level collocations is that the model contains more word pairs, which means we will discover pairs we have not seen in the simple *bigram* method. For the bigrams we have seen before, we expect to get PMI values that are closer to the real distribution.

Our method works with morphological lemmas of words. We introduce a frequency cutoff[5] into our model, if a unigram has a lower frequency than this cutoff, we exclude it. In our experiments, we use a frequency cutoff of 1,000, which means the model does not consider hints that occur less than one thousand times in the corpus. Such a model takes roughly the 16,000 most frequent words in CzEng.

3.3. Dependency-level Collocations

In the sentence-level model, we considered many word pairs, many of those have nothing in common. To reduce noise, we introduce *dependency-level* collocations model, in which the considered word pairs are restricted to words between which there is a dependency relation. We define $\mathrm{count}(x,y)$ as the number of times *y* and *x* occur in the same sentence and have a dependency relation.

[4] *Arthur* will be just as related to *king* as *king* is to *Arthur*. Even though it might be feasible to extract the direction of the relation from the syntactic makeup of the collocation or its syntactic context, this falls outside of the scope of this paper. We instead choose to generalize and say that the co-occurrence of two words counts equally towards either direction regardless of context.

[5] The frequency cutoff is an important factor in the quality of the model. Setting the cutoff too high results in a model that is too general and cannot accurately target any particular word on the board. Setting the cutoff too low will result in very obscure words entering the model, which is problematic if these words fall outside the vocabulary of a player. A cutoff that is too low will also suffer from data sparsity. For example, if a word occurs only once or twice in the data, it has a high PMI value for the words it co-occurs with, even though the real distribution might be much different. In this case, the PMI value is most likely not representative of the actual distribution.

The CzEng corpus we use has already been annotated with syntactic and semantic information. The annotation is separated into the analytical layer and the tectogrammatical layer. The syntactic trees were created automatically by the Treex[6] pipeline. For our purposes, we use the tectogrammatical trees, which exclude function words so content words are related through dependency edges directly. This is very useful for our dependency-level method. We work only with tectogrammatical lemmas and ignore all special tectogrammatical nodes with lemmas starting with a hash sign (#). Additionally, we strip away information about reflexivity of verbs from the lemmas which is encoded with *_se* and *_si*.

The dependency-level collocations are bidirectional, the hint can be both the dependent as well as the head of the dependency relation. Whether the direction of the dependency relation plays a role in the quality of the model, would be an interesting direction for future research.

4. Evaluation

The task of word association is the retrieval of associated lexical items in response to a word prompt. In order to make this task more appealing to participants, we test word association by humans in the context of a word association game called *Codenames* (Chvátil, 2015).[7] The game is available in many languages, but we focus our efforts on English and Czech because these languages are well represented in our group of participants.

4.1. Codenames Board Game

The game is played in two teams of 2 or more players, each team has one spymaster and one or more agents. The game board consists of 25 cards with a word written on it. There are nine cards that belong to the team that starts the game, eight that belong to the opposing team, seven neutral cards and one assassin card, which loses the game for the team that selects it. Both spymasters get to see which cards belong to which team, but the agents do not. Each turn one of the spymasters gives a hint to their agents for one or more cards that belong to their team. The spymaster also gives a number that signals to how many of their own cards the hint is related. The agents then proceed to guess cards until they select one that does not belong to their team or they voluntarily end their turn because they do not see any more cards that are related to the hints that their spymaster has given.

The goal of the game is to turn over all of the cards that belong to your team. As a spymaster, you help achieve this goal by giving hints to your agents that are associated with your own cards and unambiguous as possible. As an agent, you will

[6] http://ufal.mff.cuni.cz/treex

[7] https://czechgames.com/en/codenames/

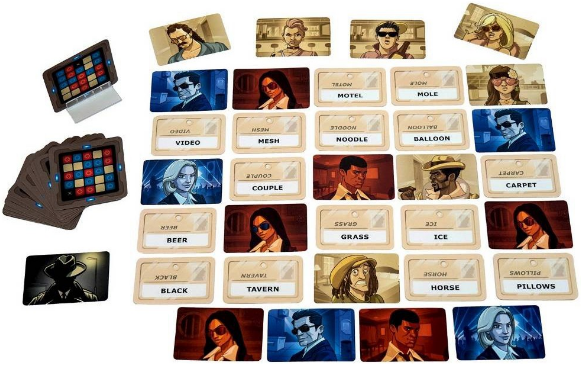


Figure 1. A regular game of Codenames

try to turn over as many cards of your own team as possible using the hints given by your spymaster, without selecting any cards that do not belong to your team and avoiding the assassin at all cost. The game ends when either team has turned over all of their cards, in which case that team wins, or one of the teams has selected the assassin in which case that team loses.

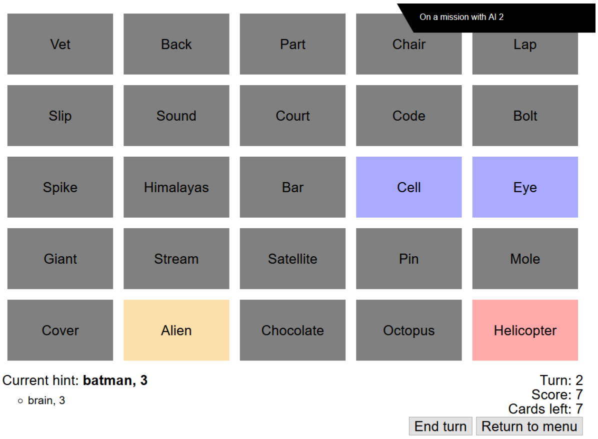
A regular game of Codenames along with the board pieces used to play the game can be seen in Figure 1. The blue and red spy cards are used to cover words that have been selected during the game. The card in the top left corner is visible only to the spymasters and shows which cards belong to which team. There are some restrictions to the hint that the spymaster can give. The hint has to be one word and cannot be any morphologically related form of a visible word on the board. For example, if one of the words on the table is *fly* and it has not been selected yet, it disallows hints such as *fly*, *flown*, *flight* and *butterfly*.

The aim of the spymaster is to provide hints that are related to the cards belonging to their team. When playing the game with other people, it can be quite challenging to give a good hint for multiple words, say *mozart, 3*, and even more so to correctly guess *symphony*, *concert* and *piano* when you get such a hint as an agent.

4.2. Our Implementation of the Game

There also exists a two-player variant, which is detailed in the rule book of the board game.[8] For our purposes, we adapt this two-player version into a version for one human player (the agent) and a computer (the spymaster) who gives hints to the human player. The game is made more regular by putting the computer and the player on the same team and introducing a dummy team that opposes them. The

[8] https://czechgames.com/files/rules/codenames-rules-en.pdf



Vet	Back	Part	Chair	Lap
Slip	Sound	Court	Code	Bolt
Spike	Himalayas	Bar	Cell	Eye
Giant	Stream	Satellite	Pin	Mole
Cover	Alien	Chocolate	Octopus	Helicopter

Current hint: **batman, 3**

- brain, 3

Turn: 2
Score: 7
Cards left: 7

Figure 2. Screenshot of our online implementation of the game.

dummy team turns over one of their own cards at random during their turn and then passes the game back to the player.

Our implementation of the game as a tool for evaluating word-associations is available online[9] and the code is available on GitHub[10]. It includes code for running the web application, generating models and the anonymized data from our experiments. A major focus while designing the application was to increase the number of games played, so we can collect more data for evaluation.

A screenshot of our implementation is given in Figure 2. We use blue for the players own cards, red for the enemy team's cards, yellow for the neutral cards and black to indicate the assassin. Gray cards have not been selected by the player yet and can be of any type. The status bar on the top right shows the name of the AI that generates the hints. On the bottom left the player can see the current hint as well as a history of the previous hints provided by the AI. On the bottom right we show the current turn, the score that the player would achieve if they guessed all of their cards in this turn and the number of own cards that the player has left. "End turn" allows the player to end their turn without selecting an incorrect card.

4.3. Evaluation Metrics

We establish micro-averaged precision, recall and f-score for measuring the quality of individual models tested on the Codenames game. The true positives are the cards

[9] `https://ufal.mff.cuni.cz/david-marecek/codenames/`

[10] `https://github.com/mderijk/codenames`

clicked by the player which were their own, false positives are the cards that the player clicked which were not their own, and the false negatives are the cards that the player should have clicked but didn't.

We also provide a baseline win rate for completeness. Although this statistic is useful, it also demands much more data to arrive at accurate results. As such, it is less suitable for our purposes since gathering enough data to compute this measure would require a lot of time. We therefore do not consider this metric for our models. By considering the decisions taken by the player instead of tallying how many games were won and lost, we are able to provide more accurate metrics and evaluate models using fewer games.

5. Aggregation of Scores

The three methods proposed in Section 3 give us a similarity score: a number that describes how much two given words are associated with each other. However, for the purpose of evaluation, we will need to generate hints that are associated potentially with multiple words at once and not associated with the enemy words. Therefore, we have to find a way to aggregate these similarity scores into one number, which we will call the **aggregate score**. We also refer to aggregation as *weighting* because of the weights that are applied to the similarity scores when they are fed to the aggregation method.

Our general strategy is to split the similarity scores into four groups: own, enemy, neutral and assassin. Each containing the similarity scores for the hint and a word from the player's own cards, the enemy's cards, the neutral cards or the assassin cards, respectively. Another categorization we make is a more simple one. We divide the words on the board into positive and negative words, where the player's own words are the positive words while all other words are the negative words.

The simplest aggregation method is to sum the similarity scores for all the positive words. This works well because the more related a word is to the hint the more it contributes to the aggregated score. The problem with this approach is that it does not take the negative scores into account at all. For a hint with 3 positive words with a score of 10, there might also be a negative word with a score of 15. This is problematic because a player will be very likely to choose the negative word over one of the positive words, thus making an incorrect decision and wasting a turn. This last point reveals an important point in the decision-making process: selecting certain types of cards is worse than others. Thus, when choosing a hint, we should also factor the type of card into the equation.

For this purpose, we introduce **weights**. These weights consist of four integers, one for player, enemy, neutral and assassin scores. The similarity scores for each category are multiplied by their category-specific weight before they are fed to the aggregation method.

$$similarity\_score(hint, word) * weight(category(word))$$

For example, if the model is considering the hint *apple*, and there is the positive word *pie*, which has a PMI score of 14.051 for *apple*, and a negative word *tasty*, let's say the assassin, with a similarity score of 9.468. Now we apply the weights, say 1 for positive words, and 2 for negative words. The similarity score of the assassin in relation to the hint becomes 18.936 instead of 9.468, and the score for the positive word is still 14.051. The aggregation method can then decide that $18.936 > 14.051$ and reject the hint because the risk that the player will select the assassin is too high.

In the next sections, we show several weighting schemes which can be applied to the relatedness scores of the models (PMI and cosine distance) to find the best hint in a game. The different weighting schemes are different functions for aggregating the similarity scores of the words in a game given a potential hint. They give different priorities to different types of cards. All our models use the same weights: the positive and the neutral words are multiplied by 1, the negative words by 1.2, and the assassin word is multiplied by 2. We would really like to avoid the assassin because this ends the game immediately, hurting our recall considerably. We also want to avoid clicking enemy cards because it costs the player a turn. Clicking a neutral card is not as bad because it is similar to getting a new hint by ending the turn and we also get to eliminate another card from play without penalty.

5.1. Combined Maximum

To calculate CombinedMax we first determine a threshold by taking the maximum similarity score from the list of negative words N. We then sum the scores from the list of positive words P that are above the threshold to get the aggregate score.

$$CombinedMax(P, N) = \sum_{x}^{P} \begin{cases} x & \text{if } x \geq \max(N) \\ 0 & \text{otherwise} \end{cases}$$

This way a hint only scores well if it relates to many words that are more similar to the hint than the most similar negative word. This implicit negative threshold is the most distinctive feature of this model.

This method is very sensitive to the weights we apply to the negative words. If we set the weights too high, this method is very good at finding the blind spots of a model. For example, for a collocations model, it might find a hint for which there is one positive word with a high PMI score while the rest of the scores are zero. The reason that this happens is that when the weights are high, there are very few positive words that can cross the implicit negative threshold if there is a negative word with a score higher than zero. Therefore, the chance that the model will find a hint for which all words have a PMI value of zero except for one positive word, is very high.

5.2. Mean Difference

The Mean Difference method simply takes the difference between the averaged score of the positive cards P and the averaged score of the negative cards N.

$$\mathit{MeanDiff}(P, N) = \frac{\sum_{x}^{P} x}{|P|} - \frac{\sum_{y}^{N} y}{|N|}$$

The problem with Mean Difference arises when there is a high variance within one of the classes. For instance, it does not account for situations where there is one negative word that has a really high similarity score with the hint and overshadows the positive words leading the player to click an incorrect card. As such, it is not as good at the start of the game when the mean can obscure negative words with high similarity if it is surrounded by many negative words with low similarity to the hint. Near the end of the game, this method becomes much better because each peak in similarity of individual words is reflected more strongly in the mean of either class.

5.3. Most Similar

This weighting method is different from the others since it does not aggregate the similarity scores of the positive and negative words. Rather, the most_similar method in Gensim[11] (Řehůřek and Sojka, 2010) works by performing vector arithmetic, adding the embeddings of the positive words to each other and subtracting the negative vectors. The method then returns the words whose vectors are closest to the resulting point.

This method performs well in targeting positive words. However, because it subtracts negative vectors and literally ”stays away” from the negative words, it can easily suffer from one simple mistake: including too many negative words. In other words, it assigns too much weight to negative words and starts generating hints that are specifically not referring to negative words, rather than providing hints that are similar to the positive words. To resolve this issue we let it take only the assassin word into account for the negative words.

5.4. Top-n

The top-n ($n \in 1, 2, 3$) methods are an adaptation of the CombinedMax function. The formula is the same, except for the fact that P is restricted to the n highest values in P. The distributional characteristics of these functions are very interesting because we have some control over its behaviour by setting n. If we take the Top-1 method, we will simply get the hint with the highest similarity score among all pairs of hint and

[11] https://radimrehurek.com/gensim/

target words. This results in hints with very large similarity scores which are usually highly associative. The Top-2 method is generally more mixed, with one hint with a high similarity score and one hint with a moderate similarity score. And if we look at the Top-3 method, we often get three words with moderately high similarity scores. Of course, there is a lot of variation depending on the number of words still on the board. The top-n methods are still similar to CombinedMax in the sense that they only have an upper bound n and no lower bound. A top-n is also allowed to give hints for less than n words.

6. Results

In this section, we show the evaluation results of the proposed word association models and aggregation methods. Since we used an iterative approach in the design of our methods, we dedicate one section to each iteration of models. In Section 6.1 we establish a baseline for our models by Monte Carlo simulation. Then we analyze the results of the first test run in Section 6.2 and discuss the improvements we made to our models in Section 6.3. The Top-n models are combined in Section 6.4. Finally, we discuss the performance of a combination of a word embedding and collocation model in Section 6.5.

6.1. Random Baseline

The baseline is set by a scenario in which hints do not provide any help to the player whatsoever, which is equivalent to the situation where there are no hints at all and cards are chosen randomly by the player.[12] We perform a Monte Carlo simulation of playing the game by repeatedly selecting cards at random. We simulate 10 million games in this way, from which we obtain the results displayed in the first row of Table 1. The baseline for the win rate, the chance to win the game by selecting cards at random, is 0.39%. This is a very low number, on average this means the player wins only one game out of 257 games.

If the generated hints provide any semantic meaning related to the player's words more so than to the other team's words, we would expect the average win rate to be higher than the baseline. The same can be said for precision, recall and f-score.

6.2. Initial Models

The results for our initial models are shown in Table 1. All of our models perform above the baseline, which means they are better than random chance. Globally, it seems that better results were obtained for Czech. We hypothesize that this is caused

[12] The end turn button that is present in the game is not modelled as a possible action because a player clicking cards randomly does not gain any additional information from getting a new hint, while the opposing team does have the opportunity of turning over an additional card.

Setup		Player decisions					
		CZ			EN		
	Aggregation	P	R	F_1	P	R	F_1
Random baseline							
	—	0.389	0.339	0.362	0.389	0.339	0.362
Word embeddings							
	MostSimilar	0.616	**0.753**	**0.677**	0.563	**0.607**	0.584
	CombinedMax	0.558	0.578	0.568	**0.567**	0.606	**0.586**
Sentence-level collocations							
	CombinedMax	0.507	0.490	0.498	0.500	0.466	0.482
Dependency-level col.							
	CombinedMax	**0.629**	0.654	0.641	0.547	0.497	0.521
	MeanDiff	0.575	0.636	0.604	0.546	0.544	0.545

Table 1. Micro-averaged precision, recall and f-score for our initial models.

by the fact that our group of English speaking players consists mostly of second language learners, while most of the players for Czech were native speakers.

Comparing the f-scores for Czech, we can see that the best method is the WordEmbeddings with MostSimilar aggregation (0.677), followed by the two Dependency level collocations models (0.641 and 0.604), and then the WordEmbeddings model with CombinedMax aggregation (0.568). For English, the best models are both aggregations of WordEmbeddings (0.584, 0.586), followed by Dependency-level collocations (0.521, 0.545).

The Sentence-level collocations were outperformed by the other two models for both languages with f-scores of 0.498 and 0.482. Even though we expected that the lack of data for the dependency model might hurt its performance, it seems that the constraints on the word pairs lead to more accurate results. In the following evaluations, we do not continue with the Sentence-level collocation model, since it did not prove to be promising. Although it contains many more word pairs, dependencies seem to capture more accurate relations between words thus producing better hints.

6.3. Improved Models

In this section, we improve our dependency and word embedding models by introducing new aggregation methods. We start with the Top-1 aggregation method,

Setup		**Player decisions**					
		CZ			**EN**		
	Aggregation	P	R	F_1	P	R	F_1
Random baseline							
	—	0.389	0.339	0.362	0.389	0.339	0.362
Sentence-level collocations							
	CombinedMax	0.507	0.490	0.498	0.500	0.466	0.482
Dependency-level col.							
	CombinedMax	0.629	0.654	0.641	0.547	0.497	0.521
	MeanDiff	0.575	0.636	0.604	0.546	0.544	0.545
	Top-1	**0.722**	0.633	0.675	**0.693**	0.678	**0.685**
	Top-2	0.621	**0.778**	**0.691**	0.646	**0.711**	0.677
	Top-3	0.552	0.644	0.595	0.655	0.611	0.632
Word embeddings							
	MostSimilar	0.616	0.753	0.677	0.563	0.607	0.584
	CombinedMax	0.558	0.578	0.568	0.567	0.606	0.586
	Top-1	**0.789**	**0.789**	**0.789**	**0.768**	0.735	**0.751**
	Top-2	0.608	0.690	0.647	0.614	0.735	0.669
	Top-3	0.574	0.626	0.599	0.667	**0.786**	0.722

Table 2. Micro-averaged precision, recall and f-score for our improved models.

which always tries to find a hint for only one word.[13] At this point, we introduce a number that shows how many target words the hint relates. We show this number to the player together with the hint for all models other than the initial models evaluated in Section 6.2. For the Top-2 and Top-3 methods, it might be the case that they do not manage to find a hint for the intended amount of words. In such cases, the number provided by the model will reflect the actual number of words that it has managed to target with the given hint. The results are shown in Table 2.

During testing, we notice a major effect of knowing the number of words that the system is hinting at. The player now knows when they have exhausted a hint and can stop using it. If the hint was for only one word and the player has selected this card, they will now press the end turn button to gain a new hint whereas previously they might have continued guessing using the same hint which would have been similar to random guessing.

[13] It is not possible to win the game this way through association alone because the maximum number of hints a player can get is 8, which can be achieved by manually ending the turn 7 times in a row. Even though this is not as fun for our participants, it provides a useful baseline.

For the Dependency model, we can see that the Top-1 and Top-2 models provide a significant improvement over the previous models for both Czech and English. The Top-3 model outperforms the original CombinedMax only for English, while it is significantly worse for Czech. The Top-2 dependency model achieves a noteworthy recall compared to the other dependency models. In this case, high recall means that players on average get much closer to turning over all of their cards and winning the game. The Top-1 models achieve the highest precision across the board. This is not surprising, it is easy to give a good hint for one word, but much harder to give a good hint for two or more words and still have the player guess both of them.

For the WordEmbedding models, we see that the Top1 model performed best for both Czech and English. The Czech Top-3 model performs poorly similar to the Dependency models, however, the English Top-3 model performs very well. The Top-2 word embedding models are considerably worse than their Top-1 counterparts, contrary to what we see for the dependency models.

We observe across all models that the Top-1 model has higher precision than recall and for the Top-2 and Top-3 models this relation swaps and the recall is higher than the precision. The only anomaly is the English Top-3. Curiously, its precision is much higher than for the Czech model.

6.4. Threshold Models

We would like to build a model that can give hints for 1, 2, and 3 words depending on the situation. Naturally, we would like to prioritize hints that target more words, so we propose a threshold model which gives hints using the Top-3 model while these hints score above some threshold and switches to the Top-2 model when no hint from the Top-3 model passes this threshold anymore. Similarly, it will switch to the Top-1 model if the score threshold for the Top-2 model can no longer be surpassed by any hint. In order to build this model, we will first need to determine adequate thresholds.

To determine these thresholds we studied the decisions made by players playing with the Top-1, Top-2, and Top-3 dependency models. For each method, we manually select a threshold value that reasonably separated hinted positive cards from the others.

We create the threshold models Top-N for both Dependency-level collocations and Word embeddings. A model consists of three submodels which we have already tested individually so we can see if there is an improvement. Hints are chosen by querying the Top-3, Top-2 and Top-1 models in that order and selecting the first hint from the model that passes its respective threshold, defaulting to the Top-1 model if none of the thresholds is passed.

Table 3 compares the results of the Top-N models and individual models. The Dependency model performed very poorly, it did not manage to outperform even the worst individual model, which was the Top-3 model. The performance of the English model is exceptionally bad when contrasted with the performance of its worst

		Player decisions					
Setup		**CZ**			**EN**		
	Aggregation	P	R	F_1	P	R	F_1
Random baseline							
	—	0.389	0.339	0.362	0.389	0.339	0.362
Dependency-level col.							
	Top-1	**0.722**	0.633	0.675	**0.693**	0.678	**0.685**
	Top-2	0.621	**0.778**	**0.691**	0.646	**0.711**	0.677
	Top-3	0.552	0.644	0.595	0.655	0.611	0.632
	Top-N	0.598	0.585	0.591	0.570	0.476	0.519
Word embeddings							
	Top-1	**0.789**	**0.789**	**0.789**	**0.768**	0.735	**0.751**
	Top-2	0.608	0.690	0.647	0.614	0.735	0.669
	Top-3	0.574	0.626	0.599	0.667	**0.786**	0.722
	Top-N	0.673	0.623	0.647	0.738	0.679	0.707

Table 3. Micro-averaged precision, recall and f-score for the threshold models.

submodel and performs much worse than the Czech model in this regard. We hypothesize that the lower threshold for the English Top-3 model has contributed significantly to this poor performance. For the Czech model, there was a much smaller gap between the threshold of the Top-3 and Top-2 model. In addition, we can say that the threshold method has not had the desired effect. While we would expect that the Top-N model would perform equally or better than the worst performing model, our English dependency model performed much worse than the worst individual model.

For the WordEmbeddings model, the picture looks slightly better. The Top-N models perform worse than the best individual models, but better than the worst individual model. While this performance is certainly better than that of the Top-N Dependency model, it does not improve over the best individual model in any way. When we look at Figure 3 we see that the threshold model did not prevent the player from selecting cards with low similarity scores. The number of positive cards selected which were not hinted at in the current turn is much higher, which explains why the model has higher precision than the Top-2 and Top-3 models. Therefore, we conclude that the threshold system successfully improves the precision of the model. However, this happened at the cost of the recall. And it still performs worse than the Top1 model across all statistics.

All Top-N models suffered in terms of recall when compared to the individual models. None of them has higher recall than the lowest recall of any of their submod-

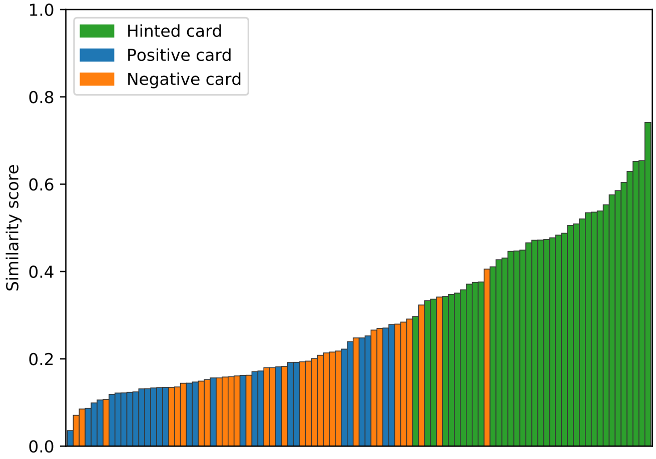


Figure 3. Similarity scores for each card clicked by players across several games for the Czech Top-N word embedding model

els. Precision, on the other hand, increased considerably in comparison to the Top-3 and Top-2 models.

The threshold model did not live up to expectations because it did not prevent the player from clicking on cards with low similarity scores in regards to the hint. We suspect that the thresholds we selected were far from optimal and a different ensemble approach might achieve better results. Finding a good way to combine the Top-1, Top-2, and Top-3 methods to achieve the same or better performance than either of the individual methods is an interesting direction for future research.

6.5. Combined Models

Lastly, we would like to test a model that combines both the dependency collocations model and the word embedding model. Since the threshold system turned out to be a poor ensembling method, we have to consider a new way in which we can combine our models. One method is to find a mapping between PMI values and cosine similarity. However, one of these measures is normalized and the other is not and their scales are radically different, so this relationship can be hard to find through trial-and-error and is in the worst case non-linear. Instead, we choose to perform ensembling through mutual agreement, where we let both models predict hints until

	Player decisions					
Setup	**CZ**			**EN**		
Aggregation	P	R	F_1	P	R	F_1
Random						
Baseline	0.389	0.339	0.362	0.389	0.339	0.362
Dependency-level col.						
Top-N	0.598	0.585	0.591	0.570	0.476	0.519
Word embeddings						
Top-N	0.673	0.623	0.647	0.738	0.679	0.707
Dependency col. & Word emb.						
Top-N combined	0.642	0.678	0.659	0.711	0.697	0.704

Table 4. Micro-averaged precision, recall and f-score for the combined Top-N dependency and word embedding model.

one of the models gives a hint that the other model has also predicted for the current board state.

We expect an ensemble model that combines word embeddings and collocations to perform better than the individual models since they both model different types of association. Word embeddings capture similarity while collocations usually capture other types of relations. Combining the best of both models should lead to better results.

We test an ensemble model that combines the Top-N dependency and Top-N word embedding models described in Section 6.4. In Table 4 we can see the results of combining dependency and word embedding models by finding hints through mutual agreement between models. The combined model performed similarly to the best models included in them with an f-score of 0.659 for Czech and 0.704 for English. The f-score of the Top-N word embedding model is slightly lower for Czech (0.647) and slightly higher for English (0.707).

Although these results are promising, they do not significantly improve the results of the models they combine. The model is successful at mimicking the performance of the best submodel, but it does not select the best hint from either model depending on what is best in a given situation. This is due to the ensembling method used. As such, more research on good ensembling methods is needed to find models that do improve above the performance of their internal parts.

In Table 5 we show the number of games played and the number of decisions made for each model. The number of decisions for a model is the sum of all the cards clicked by players in the games played with that model.

		Player decisions			
Setup		**CZ**		**EN**	
	Aggregation	#G	#D	#G	#D
Sentence-level collocations					
	CombinedMax	17	148	62	520
Dependency-level col.					
	CombinedMax	17	159	68	556
	MeanDiff	18	179	67	601
	Top1	10	79	10	88
	Top2	11	124	10	99
	Top3	10	105	10	84
	TopN	10	92	10	86
Word embeddings					
	most_similar	22	242	65	630
	CombinedMax	25	233	77	741
	Top1	10	90	13	112
	Top2	14	143	13	140
	Top3	11	108	13	138
	TopN	10	98	11	103
Dep. collocations & WE					
	TopN - mutual	10	95	11	97

Table 5. The number of games played (#G) and the number of decisions (#D) made for all models tested.

7. Conclusions

We have provided both a theoretical and practical framework for the evaluation of computational models of word associations. We started out by establishing a baseline for the task of Codenames with a single human player. After this, we explored several methods all of which performed well above the baseline. The restriction on syntactically dependent words proved a definite improvement over broad sentence-level word pairs for the collocation model for both evaluated languages.

Large improvements to our model were made by aggregating the similarity scores of the words on the board and weighting them more cleverly. Our best Dependency models achieved an f-score of 0.691 for Czech and 0.685 for English. The Word Embedding models based on the same aggregation technique, in turn, outclassed these models with f-scores of 0.789 and 0.751 for Czech and English, respectively. The model that got closest to helping the player turn over all their cards, was the Top-2 Dependency model for Czech with a recall of 0.778. For English, the best model in this regard was the Top-3 Word Embeddings model with a recall of 0.786.

We made several attempts to build ensemble models that combine the best performing models to boost their performance. We were not successful in this regard, our Top-N dependency model achieved f-scores of 0.591 and 0.519 for Czech and English respectively. The Top-N word embeddings performed better, with an f-score of 0.647 for Czech and 0.707 for English, but neither outperformed the best individual Top-n model for their respective language. A final attempt at combining dependency and word embedding models by finding hints through mutual agreement between models performed similarly to the best models included in them with an f-score of 0.659 for Czech and 0.704 for English. Although these results are promising, we believe that many better ensembling methods still remain.

We have shown that both dependency-level collocation models and word embedding models can provide hints of considerable quality, given the right constraints. Dependency models manage to capture several types of relations between words which the player is able to pick up on, while the word embedding models excel at finding semantically similar hints.

8. Future Work

We have provided an overview of only the most basic methods and we believe that many improvements can still be made to achieve better performance on the Codenames word association task. For example by finding better ensemble methods to combine models that give hints for a different number of words, as well as successfully combining models of different types such as collocation and word embedding models.

The methods we use are themselves simple baselines for the technique that they are based on. There exist many more measures of association other than pointwise mutual information (see Pecina (2010) for an extensive list of such association measures) and there have been many improvements in recent years over the fastText word embeddings that we tested, many of which might surpass our best word embedding models when compared. This could be a fruitful direction for future research.

The same framework can be used to analyze the effect of time taken between receiving the word prompt and making a decision. We did not incorporate any timing mechanism in our application, so it is not possible to extract this type of information from our dataset. However, it is easy to modify the application and record this data as well, so this is nonetheless an interesting avenue for future work.

While this paper was mainly focused on the computational side of word association, it must be noted that a human baseline for the Codenames word association task would be very useful to give more context to the results achieved on this task. Similarly, comparing the predictions made by the models to human-level word associations would be a useful direction in this area.

Acknowledgements

This work has been supported by the grant 18-02196S of the Czech Science Foundation. We would also like to thank our testers, who helped us with evaluating the proposed models by playing the online game and thank the anonymous reviewers for many insightful comments and suggestions.

Bibliography

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. ISSN 2307-387X. doi: 10.1162/tacl_a_00051.

Bojar, Ondřej, Ondřej Dušek, Tom Kocmi, Jindřich Libovický, Michal Novák, Martin Popel, Roman Sudarikov, and Dušan Variš. CzEng 1.6: Enlarged Czech-English Parallel Corpus with Processing Tools Dockered. In Sojka, Petr, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, *Text, Speech, and Dialogue: 19th International Conference, TSD 2016*, number 9924 in Lecture Notes in Computer Science, pages 231–238, Cham / Heidelberg / New York / Dordrecht / London, 2016. Masaryk University, Springer International Publishing. ISBN 978-3-319-45509-9. doi: 10.1007/978-3-319-45510-5_27.

Chvátil, Vlaada. Codenames. Czech Games Edition, 2015.

De Deyne, Simon, Danielle Navarro, Amy Perfors, Marc Brysbaert, and Gert Storms. The “Small World of Words” English word association norms for over 12,000 cue words. *Behavior Research Methods*, 51, 10 2018. doi: 10.3758/s13428-018-1115-7.

Fellbaum, Christiane, editor. *WordNet: an electronic lexical database*. MIT Press, 1998. doi: 10.7551/mitpress/7287.001.0001.

Gough, Harrison G. Studying creativity by means of word association tests. *Journal of Applied Psychology*, 61(3):348–353, 1976. doi: 10.1037/0021-9010.61.3.348.

Jung, Carl G. The association method. *The American journal of psychology*, 21(2):219–269, 1910. doi: 10.2307/1413002.

Levy, Omer and Yoav Goldberg. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 302–308, 2014. doi: 10.3115/v1/P14-2050.

Liu, Yang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Topical word embeddings. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

Morkovkin, V. V. *Ideographic Dictionaries*. 1970.

Nelson, Douglas L., Cathy L. McEvoy, and Thomas A. Schreiber. The University of South Florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, 36(3):402–407, 2004. doi: 10.3758/BF03195588.

Obrtlík, Petr. Computer as an intelligent partner in the word-association game codenames. Master's thesis, Brno University of Technology, Brno, 2018.

Pecina, Pavel. Lexical association measures and collocation extraction. *Language resources and evaluation*, 44(1-2):137–158, 2010. doi: 10.1007/s10579-009-9101-4.

Řehůřek, Radim and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. URL `http://is.muni.cz/publication/884893/en`.

Thawani, Avijit, Biplav Srivastava, and Anil Singh. SWOW-8500: Word Association task for Intrinsic Evaluation of Word Embeddings. In *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pages 43–51, Minneapolis, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-2006. URL `https://www.aclweb.org/anthology/W19-2006`.

Address for correspondence:
David Mareček
marecek@ufal.mff.cuni.cz
Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics,
Charles University, Malostranské náměstí 25, 118 00 Praha, Czechia

The Prague Bulletin of Mathematical Linguistics
NUMBER 114 APRIL 2020

INSTRUCTIONS FOR AUTHORS

Manuscripts are welcome provided that they have not yet been published elsewhere and that they bring some interesting and new insights contributing to the broad field of computational linguistics in any of its aspects, or of linguistic theory. The submitted articles may be:

- long articles with completed, wide-impact research results both theoretical and practical, and/or new formalisms for linguistic analysis and their implementation and application on linguistic data sets, or
- short or long articles that are abstracts or extracts of Master's and PhD thesis, with the most interesting and/or promising results described. Also
- short or long articles looking forward that base their views on proper and deep analysis of the current situation in various subjects within the field are invited, as well as
- short articles about current advanced research of both theoretical and applied nature, with very specific (and perhaps narrow, but well-defined) target goal in all areas of language and speech processing, to give the opportunity to junior researchers to publish as soon as possible;
- short articles that contain contraversing, polemic or otherwise unusual views, supported by some experimental evidence but not necessarily evaluated in the usual sense are also welcome.

The recommended length of long article is 12–30 pages and of short paper is 6–15 pages.

The copyright of papers accepted for publication remains with the author. The editors reserve the right to make editorial revisions but these revisions and changes have to be approved by the author(s). Book reviews and short book notices are also appreciated.

The manuscripts are reviewed by 2 independent reviewers, at least one of them being a member of the international Editorial Board.

Authors receive a printed copy of the relevant issue of the PBML together with the original pdf files.

The guidelines for the technical shape of the contributions are found on the web site `http://ufal.mff.cuni.cz/pbml`. If there are any technical problems, please contact the editorial staff at `pbml@ufal.mff.cuni.cz`.