SLICE: Supersense-based Lightweight Interpretable Contextual Embeddings

Cindy Aloui, Carlos Ramisch, Alexis Nasr, Lucie Barque

SLICE: Supersense-based Lightweight Interpretable Contextual Embeddings

Cindy Aloui[1] and **Carlos Ramisch**[1] and **Alexis Nasr**[1] and **Lucie Barque**[2]
[1] Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
[2] Université Sorbonne Paris Nord, LLF, Paris, France
cindy.aloui@gmail.com alexis.nasr@lis-lab.fr
carlos.ramisch@lis-lab.fr lucie.barque@univ-paris13.fr

Abstract

Contextualised embeddings such as BERT have become *de facto* state-of-the-art references in many NLP applications, thanks to their impressive performances. However, their opaqueness makes it hard to interpret their behaviour. *SLICE* is a hybrid model that combines supersense labels with contextual embeddings. We introduce a weakly supervised method to learn interpretable embeddings from raw corpora and small lists of seed words. Our model is able to represent both a word and its context as embeddings into the same compact space, whose dimensions correspond to interpretable supersenses. We assess the model in a task of supersense tagging for French nouns. The little amount of supervision required makes it particularly well suited for low-resourced scenarios. Thanks to its interpretability, we perform linguistic analyses about the predicted supersenses in terms of input word and context representations.

1 Introduction

The form-meaning association relating words to their senses is a fundamental component of human languages. Hence, lexical semantics, that is, the representation of the *meaning of words*, is an important research topic in computational linguistics. Processing word meaning is essential for the (compositional) interpretation of larger units such as phrases and sentences. Therefore, computational lexical semantics is, explicitly or implicitly, at the core of higher-level NLP tasks such as textual understanding, information extraction, and automatic summarisation.

Much effort has been put in the manual and semi-automatic construction of resources encoding lexical semantics (i.e., word meaning). These include semantic lexicons with inventories of possible senses that lexical units can assume (e.g., Wordnet) and sense-annotated corpora specifying which of these senses are employed in context (e.g., SemCor). Alternatively, real-numbered vectors can encode contextual co-occurrence, acting as a proxy for a lexical unit's semantics. This principle has guided the development of numerous distributional semantic models, that is, semantic vector representations inferred from corpus co-occurrences, e.g., Landauer and Dumais (1997). Advances in neural networks shifted the focus of computational semantics to representation learning, so as to obtain vectors as by-products of neural networks (Mikolov et al., 2013), In this booming field, a myriad of models have emerged, efficiently learned from corpora, benefiting from high-performance neural architectures and libraries. Thus, vector representations, rebranded *word embeddings*, have become the dominant technique to represent lexical units, at the core of state-of-the-art neural approaches.

Traditional *static* embeddings, such as word2vec and Fasttext, assume that each word's meaning can be represented as a single vector, independently of its context. While generic and reusable, these models usually conflate the different meanings of a given unit into a single vector (Camacho-Collados and Pilehvar, 2018). *Contextual* models, such as ELMo, GPT-2, BERT, and their variants, encode each word's occurrence as a context-dependent vector, assuming that each context corresponds to a different sense (Yarowsky, 1993). In short, while static models create one generic embedding per lexical unit, contextual

models provide a fine-grained distinct representation for each occurrence. Both models, but especially the latter, are increasingly complex and opaque (Rogers et al., 2020), requiring advanced techniques to help humans understand their strengths and limitations (Jawahar et al., 2019; Serrano and Smith, 2019).

Given this landscape, we introduce *SLICE*: an alternative semantic model which constitutes a trade-off between static, interpretable symbolic senses and contextual word embeddings. We propose a **weakly supervised technique to build dense low-dimensional embeddings** whose dimensions represent coarse-grained semantic classes i.e., supersenses such as ANIMATE ENTITY and NATURAL OBJECT (Sec. 3). **Our lightweight model embeds both lexical units and their contexts into the same semantic space.** Thus, words and their contexts are represented as two compact vectors of **directly interpretable scores, one per supersense**, automatically learned from a non annotated corpus. Our embeddings are assessed in a supersense tagging setting (Sec. 4). Thanks to the model's interpretability, we are able to perform a **rich linguistic analysis of the results**, providing insights to understand the model's predictions (Sec. 5).

2 Related Work

Our work is positioned at the crossroads of word and sense embeddings, interpretable semantic representations, and weakly supervised semantic classification. We briefly review a sample of relevant work on these topics.

Word and sense embeddings The literature on vector-space semantic representations is enormous, ranging from traditional models such as LSA (Landauer and Dumais, 1997) to sophisticated deep contextualised embeddings such as BERT (Devlin et al., 2018). Although techniques are being constantly improved, the main principle is stable across models: vectors represent a word's usage (and meaning) based on its distributional context (Harris, 1954). Embeddings have become commonplace in NLP, as they naturally represent input (words) in state-of-the-art neural models. Although they can be randomly initialised and learned, unsupervised pre-training on raw corpora is common (Turian et al., 2010).

Embeddings can be pre-trained as by-products of predictive neural language models (Mikolov et al., 2013), by factorisation of the co-occurrence matrix (Landauer and Dumais, 1997; Pennington et al., 2014), etc. Sub-lexical units (character n-grams) address linguistic variability, e.g., due to rich morphology, non-standard text, and out-of-vocabulary forms (Bojanowski et al., 2017). Most of the models prior to 2018 are *static*, assuming a single vector per word. These models suffer from meaning conflation, i.e., a single vector is created for ambiguous units, ignoring polysemous and multi-facet words.

Advances in neural networks triggered the development of *contextual* embeddings, with representations conditioned on the surrounding words. They can be obtained using stacked recurrent layers as in ELMo (Peters et al., 2018), or attention-based transformers as in BERT (Devlin et al., 2018) and GPT-2 (Radford et al., 2018). In addition to their outstanding performances, these models address meaning conflation: contexts correspond to (slightly) different senses and are modelled with a custom embeddings. On the downside, they are computationally heavy and opaque (Rogers et al., 2020), requiring sophisticated techniques such as probing to interpret predictions (Jawahar et al., 2019).

Particularly relevant to our work are *sense embeddings* (Camacho-Collados and Pilehvar, 2018), in which a lexical unit is associated to several vectors (as in contextual models), but with some generalisation across occurrences (as in static models). Unsupervised sense (multi-prototype) embeddings can be obtained by adapting the objective of the learning procedure (Neelakantan et al., 2014), or with word sense induction methods based on clustering, e.g., Panchenko et al. (2017). For interpretability, resources such as Wordnet can be used to semantically enhance static embeddings (Faruqui et al., 2015) or to learn representations for Wordnet synsets (Rothe and Schütze, 2015), supersenses (Flekova and Gurevych, 2016), or Babelnet senses (Camacho-Collados et al., 2016). Contextual models such as BERT can be enriched with supersenses, predicted jointly with masked words during training, with observed improvements in tasks requiring lexical semantics (**?**).

Interpretable semantic representations One of the most popular sense inventories in NLP is Wordnet (Miller et al., 1990), in which words are grouped into synsets and linked to each other via lexical-semantic relations (e.g., hypernymy, synonymy). For many years, the English Wordnet has been the basis of

sense-annotated corpora (Landes et al., 1998) and WSD research (Navigli, 2009). Babelnet (Navigli and Ponzetto, 2012) is a semi-automatic multilingual lexicon similar to Wordnet and also quite popular for performing WSD in languages other than English (Moro et al., 2014).

Supervised WSD relies on sense-annotated corpora specifying which of the senses in the inventory are employed in context (Pasini and Camacho-Collados, 2020), e.g., SemCor for English Wordnet (Landes et al., 1998) and Eurosense for Babelnet (Delli Bovi et al., 2017). The fine granularity of sense inventories is often criticised as unrealistic (Navigli, 2009). One alternative is to represent senses using top-level synsets in Wordnet's taxonomy (e.g., ANIMAL, EVENT), referred to as *supersenses*, reached via hypernymy relations (Ciaramita and Johnson, 2003; Schneider et al., 2016). This reduces the number of labels at the expense of missing potentially relevant distinctions, often with positive impact on downstream applications such as dependency parsing (Agirre et al., 2011) and personality profiling (Flekova and Gurevych, 2015).[1] In our evaluation, we employ the FrSemCor corpus, a French corpus in which nouns are annotated using Wordnet supersenses as semantic tags (Barque et al., 2020).

Although the set of 25 Wordnet top-level categories is quite popular, alternative representations with even coarser granularity can be useful for downstream applications (Jahan et al., 2018), such as a three-way classification of adjectives (Boleda et al., 2012) or animate vs. inanimate nouns (Øvrelid, 2006). We understand supersenses as general coarse semantic distinctions. Our set of six semantic labels is related to Wordnet supersenses, but there is not a 1:1 relation between our supersenses and Wordnet's ones.

Weakly supervised semantic classification Many models have been proposed to induce lexical semantics from raw corpora without supervision, e.g., (Lin, 1998), usually performing unsupervised WSD as a by-product. Most methods rely on distributional clustering algorithms, e.g., (Biemann and Riedl, 2013). While automatically induced word senses are hard to interpret, they may be automatically labeled, for example, using hypernym-induction patterns (Ustalov et al., 2019).

There have been several proposals to integrate interpretable representations such as supersenses with continuous (unsupervised) representations, but they often rely on annotated corpora, such as Semcor (Flekova and Gurevych, 2016) or sense inventories such as Wordnet (**?**). Our embedding learning procedure is not fully unsupervised, but uses weak supervision to bootstrap semantic classes from corpora. Typical or non-ambiguous words can be used to produce sense-annotated data, which in turn enable training classifiers for inducing lexical knowledge. This has been proposed in several studies, e.g., Mihalcea (2003), especially for polysemy pattern detection (Boleda et al., 2012), and adapted to semantic frame induction using predicate-argument pairs (Jauhar and Hovy, 2017).

The method of Thelen and Riloff (2002) is similar to ours. They learn representations for six coarse supersenses using pattern-based bootstrapping based on a small list of seed words. The features used to learn senses are based on lexical patterns, syntactic co-occurrence, web queries, etc. (Qadir and Riloff, 2012). Instead of focusing on the features, our approach is more in line with current neural methods, with features learned from the data jointly with the supersense classifiers.

3 Contextual and Lexical Signatures

The heart of SLICE consists in a series of binary classifiers, one per supersense. Each classifier takes as input a context C and produces a score that indicates how likely C could be associated to a given supersense s_i. This score, noted $cs_i(C)$, is called a *context score*. A context C can be associated to a d-dimensional vector, called its *signature* $CS(C) = (cs_1(C), \ldots, cs_d(C))^T$, where d is the number of different supersenses.[2] The classifiers are also used to model the overall tendency of a word w to occur in contexts that are representative of a given supersense s_i. This information is modeled by the *lexical scores* $ls_i(w)$, computed by aggregating the context scores of all occurrences of w in a large corpus. A word w is therefore associated to a d-dimensional vector, also called its *signature*[3] $LS(w) = (ls_1(w), \ldots, ls_d(w))^T$. Such vectors can be compared to word embeddings produced by deep learning methods. The difference, however, is that each dimension of a word signature corresponds to an interpretable supersense.

[1] A comprehensive list of downstream applications for supersense tagging can be found in Flekova and Gurevych (2016).

[2] In our case, $d = 6$, but the method can be applied to any number of semantic categories $d \geq 2$.

[3] If necessary, we use the terms *lexical signature* and *context signature* to distinguish between these objects.

As stated above, SLICE relies on classifiers that themselves require, to be trained, supersense-annotated corpora. Such corpora, when they exist, usually are of limited size and do not allow to build reliable context and word signatures. This is why we propose a semi-supervised method not requiring annotated corpora, but a list of representative words for each supersense, easier and cheaper to constitute.

3.1 Outline of the Method

We use as a starting point d disjoint sets of non-ambiguous words representative of each supersense; these words are referred to as *seeds*. Seeds' occurrences are deterministically annotated with their corresponding supersenses in a corpus $\mathcal{C}$, yielding a pseudo-annotated corpus that is used to train d classifiers, one per supersense. More precisely, the method is composed of the following steps:

1. Manually compile d disjoint sets $\mathcal{S}_1 \ldots \mathcal{S}_d$ of positive examples, each containing lemmas that are prototypical of supersenses $s_1 \ldots s_d$.[4]

2. Automatically compile d sets $\mathcal{S}_1^- \ldots \mathcal{S}_d^-$ of negative examples, each containing lemmas that do not pertain to supersenses $s_1 \ldots s_d$. The set $\mathcal{S}_i^-$ is built by selecting lemmas randomly from $\cup_{j \neq i} \mathcal{S}_j$.

3. For each supersense s_i, locate in a non annotated corpus $\mathcal{C}$ all occurrences of the words whose lemmas are elements of $\mathcal{S}_i$ or $\mathcal{S}_i^-$. Words that come from $\mathcal{S}_i$ are labelled 1 and those from $\mathcal{S}_i^-$ are labelled 0. As a result, d pseudo-annotated corpora $\mathcal{C}_1 \ldots \mathcal{C}_d$ are produced.[5]

4. Train d classifiers $P_1 \ldots P_d$ respectively on $\mathcal{C}_1 \ldots \mathcal{C}_d$. The classifier P_i takes as input a context $C = (W, k)$, where $W = w_1 \ldots w_{|W|}$ is a sentence and k is the position corresponding to the pseudo-annotated word. $P_i(C)$ returns a score $0 \leq cs_i(C) \leq 1$, indicating how representative context C is of class s_i. This score is the context score mentioned above. Contexts that are representative of class s_i will have scores close to 1.

5. For each word w, extract from $\mathcal{C}$ all contexts $C_1 \ldots C_n$ in which w occurs (contexts (W, k) such that $w_k = w$) and predict scores $cs_1(C_j) \ldots cs_d(C_j)$, $1 \leq j \leq n$ with $P_1 \ldots P_d$. For each supersense s_i, all scores $cs_i(C_j)$, $1 \leq j \leq n$ are combined to form the lexical score $ls_i(w)$, which reflects the tendency of word w to appear in contexts representative of supersense s_i. Finally, w is associated to a d-dimensional vector, its lexical signature, composed of the lexical scores $ls_1(w) \ldots ls_d(w)$.

The preceding description outlines the main steps of our method, but leaves unspecified two important aspects: the nature of the classifiers P_i used to compute context scores $cs_i(c)$, and the way context scores are aggregated into lexical scores $ls_i(w)$. They are discussed in the two following sections.

3.2 Context Scores

For each supersense s_i, context scores are computed by a binary classifier P_i trained to predict the classes 0 or 1 of the positive and negative seed occurrences w_k in the pseudo-annotated corpus $\mathcal{C}_i$. The classifiers are trained on a variant of a masked language modelling task, trying to predict the pseudo-annotated supersense of the masked word based on its context. In other words, we expect them to discriminate between contexts that are representative of a given supersense (1) vs. context that are irrelevant (0).

In practice, the input of P_i is a context $C = (W, k)$. Each word $w_j \in W$ is represented as a triple (f, l, m) where f is the surface form of the word, l its lemma and m its morphological features (e.g., number=plural).[6] Each element of this triple is represented as a randomly initialised embedding of size 500 for f and l and 64 for m.

The classifiers are made of two LSTMs: a left LSTM that processes the sentence from the first word w_1 to w_{k-1}, and a right LSTM that processes the sentence backwards, from the last word $w_{|W|}$ to w_{k+1}. The hidden-state vector size of both LSTMs is equal to 300. Notice that the LSTMs ignore the pseudo-annotated word w_k. The final states of the two LSTMs are concatenated, along with the morphological

[4] Avoiding polysemous seeds is crucial to minimise the number of (inevitable) errors in automatic annotation.

[5] Sentences not containing any word in $\mathcal{S}_i \cup \mathcal{S}_i^-$ are discarded.

[6] Morphological features are represented as one-hot vectors whose positions correspond to *lists* of key=value pairs.

features of word w_k, represented as an embedding of size 64. The resulting 664-dimensional vector is fed to a multilayer perceptron (MLP) with one hidden dense layer of size 150. The output layer is of dimension 2 corresponding to classes 0 ($w_k \in \mathcal{S}_i^-$) and 1 to predict ($w_k \in \mathcal{S}_i$), with softmax activation.

The LSTMs and the subsequent dense layer form a single network trained jointly. The loss function used to train each P_i is categorical cross entropy, and the optimiser is Adam. We use a dropout of 30% to prevent overfitting, that is, for each prediction, each lemma and form in the input have a 30% probability of being masked. The size of the batches is equal to 128, and every 30,000 examples, the accuracy on the development corpus is computed. If this accuracy is the best up to now, the model is saved, and if it does not increase for the next 10 steps of 30,000 examples, training is stopped and the best model is kept.

3.3 Lexical Scores

Lexical scores $ls_i(w)$ reflect the tendency of word w to appear in contexts representative of class s_i. It is a function of the contextual scores $cs_i(C_1) \ldots cs_i(C_n)$, where $C_1 \ldots C_n$ are all the contexts in which w occurs in the corpus $\mathcal{C}$. A context C is representative of class s_i if its score $cs_i(C)$ is close to 1 and is not representative of class s_i when $cs_i(C)$ is close to 0. Intermediate scores, close to 0.5, are less informative, so their contribution to the lexical score should be lower than that of representative scores.

We use the parabolic function $h(a) = (1 - 2a)^{2p}$ to model this behaviour. It reaches its minimum value 0 in the range $[0 \ldots 1]$ for $s = 0.5$ and its maximum value 1 for $s = 1$ and $s = 0$. Parameter p controls the extent to which intermediate scores are taken into account, the higher the value of p, the less intermediate values contribute to the lexical score (in our experiments, we arbitrarily set $p = 8$ upon observation of the distribution of the predicted context scores). The lexical score $ls_i(w)$ is defined as the average of its context scores $cs_i(C_{,}j)$ weighted by $h(cs_i(C_j))$:

$$ls_i(w) = \frac{1}{\sum_{j=1}^{n} h(cs_i(c_j))} \sum_{j=1}^{n} h(cs_i(c_j)) \times cs_i(c_j)$$

4 Experimental Setup

We describe in this section the data we have used to build contextual and lexical signatures and the data we will use in the following section to evaluate our method.

Supersense Tagset To guarantee a sufficient scope (with respect to the size of the seed lists) and to simplify analysis, we grouped Wordnet supersenses into six coarser categories for our experiments on French nouns (details in Appendix A). **Animate entity** (ANI) includes nouns referring to living and animate entities, namely persons (e.g., *agriculteur* 'farmer') and animals (e.g., *chiot* 'puppy'); **Natural object** (NAT) is for nouns referring to natural entities (e.g., *étoile* 'star'), plants (e.g., *peuplier* 'poplar tree') and body parts (e.g., *hanche* 'hip'); **Manufactured object** (MAN) is composed of nouns denoting human-made or transformed entities: artifacts (e.g., *chronomètre* 'stopwatch') or built places (e.g., *isoloir* 'voting booth'); **Informational object** (INF) includes nouns denoting abstract objects having informational contents (e.g., *théorie* 'theory'), those referring to knowledge areas (e.g., *ethnologie* 'ethnology') and financial assets (e.g., *budget* 'budget'); **Dynamic situation** (DYN) gathers nouns denoting things that happen or that are carried out, like actions (e.g., *ablation* 'removal'), activities (e.g., *cyclisme* 'cycling') and events (e.g., *explosion* 'outbreak'); **Stative situation** (STA) is for nouns that denote properties (e.g., *dignité* 'dignity'), states (e.g., *endettement* 'easement') and feelings (e.g., *tendresse* 'tenderness').

Seeds We used data provided by the *Wolf*, a French lexical resource automatically built from the Princeton Wordnet (Sagot and Fišer, 2008), to draw up the six seed lists $\mathcal{S}_i$. A list of monosemous French nouns has been extracted from this resource and then we manually selected 200 nouns for each coarser category described above. For example, the seed list for the DYN class, contains nouns manually selected from the Wolf nouns having only one supersense among those that denote dynamic situations. Selecting monosemous seeds for pseudo-annotation of the corpus can bias the classifiers, which never encounters polysemous words at training time, but only in test data. However, this should not be a problem, as the we learn to classify *contexts*, not words. That is, the absence of polysemous words among the seeds should not be problematic, assuming that most polysemous words in context are disambiguated.

Corpus and Preprocessing Experiments have been conducted on the frWaC corpus, which contains about 1.6 billion words crawled from the web (Baroni et al., 2009). The corpus has been POS tagged, lemmatised and morphologically analysed by an in-house parser trained on the French corpora of Universal Dependencies (Nivre et al., 2016). The corpus is divided into 55 parts of about 1M sentences each. Part 54 is used as development corpus for early stopping, all other parts are used for training.

Positive and negative seed sets $\mathcal{S}_i$ and $\mathcal{S}_i^-$ are split into a training set (80% of the lemmas) and a development (20% of the lemmas). The training seeds are used to annotate the training corpus, while the development seeds are used to annotate the development corpus. This is a deterministic process: each occurrence of a word in $\mathcal{S}_i$ (resp. $\mathcal{S}_i^-$) is annotated as 1 (resp. 0). We artificially balance the number of training contexts in each corpus $\mathcal{C}_i$ to avoid biases related to different distributions of positive and negative examples. Given the seed lists $\mathcal{S}_i$ and $\mathcal{S}_i^-$, we count the total number of occurrences of lemmas from each list, N_i and N_i^- in $\mathcal{C}$. If $N_i < N_i^-$, all sentences containing a lemma from $\mathcal{S}_i$ are added to $\mathcal{C}_i$. Then, sentences containing lemmas from $\mathcal{S}_i^-$ are randomly added until at least N_i occurrences from $\mathcal{S}_i^-$ appear in $\mathcal{C}_i$. If $N_i^- < N_i$ the seed lists are inverted. All other sentences are discarded.

Evaluation data The FrSemCor corpus was used for evaluation (Barque et al., 2020).[7] It contains manual annotations for more than 12,000 nouns in the Sequoia Treebank, a corpus of 3,009 sentences from different sources including morphological and syntactic annotations (Candito and Seddah, 2012). Noun tokens have been annotated with 24 supersenses adapted from the Wordnet supersense tagset.[8] For this experiment, we used 7,188 annotated nouns: 5,160 have been used for training, 1,015 for development and 1,013 for evaluation.

5 Supersense Tagging

We have evaluated SLICE on a supersense tagging task because our model produces interpretable senses that can be directly compared to senses used in semantically annotated corpora (FrSemCor, in our case). Our model produces, for every word in context, a description of the context through the context signature, and a description of the word usage through its lexical signature. Comparing different ways to combine these two pieces of information is interesting from a linguistic point of view since it can lead to interesting analyses of complex linguistic phenomena such as polysemy, multi-facet nouns, and unusual contexts (e.g., manufactured objects MAN in contexts typical of animate beings ANI).

As a comparison point for the performances reached by SLICE, we have used a simple baseline, which selects for every noun occurrence its most frequent supersense (MFS) in the training corpus. When the word does not occur in the training corpus, the most frequent supersense across all words is selected. This crude method gives better results as the training corpus grows, since the coverage grows with size of the training corpus and selecting the most frequent supersense is a good heuristic (Navigli, 2009).

We also compare our model to a state-of-the-art model in other WSD tasks: a French-specific version of BERT called FlauBERT (Le et al., 2020). We use the 1024-dimensional embeddings available in FlauBERT-large as part of the HuggingFace library.[9] For each target noun, we obtain its contextualised embedding from the top layer and provide it to an MLP identical to the one described in Section 5.2. Tokenisation incompatibilities due to BPE encoding are rare (e.g., 50/1,013= occurrences in the test corpus); they are resolved by taking the noun's last subtoken before the word separator as its embedding.

In our model, the decision to tag word w in context C with a given supersense is taken based on the lexical signature of w and the context signature of C.[10] They are combined to yield a word-in-context signature $\Psi(\mathit{LS}(w), \mathit{CS}(C))$, which is also d-dimensional. The component corresponding to the highest score is selected as the predicted supersense for w in C:

$$\hat{s}(w, C) = \underset{1 \leq i \leq d}{\operatorname{argmax}} \Psi_i(\mathit{LS}(w), \mathit{CS}(C))$$

[7] https://frsemcor.github.io/FrSemCor/

[8] The Wordnet supersenses tagset, also known as *Wordnet Unique Beginners* (Miller et al., 1990), is composed of 25 nominal supersenses. Small adjustments have been made for the annotation of French nouns (Barque et al., 2020).

[9] https://huggingface.co/

[10] In practice, our experiments use a word's *lemma* signatures instead of surface forms.

α	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
acc.	51.83	54.99	58.05	61.01	62.59	64.46	65.55	66.35	66.34	65.84	64.36

Table 1: Accuracy of the linear model for different values of α on the training set.

config.	Ref	Lex	Cont	occ	ratio	LM	MLP
AAA	A	A	A	424	0.42	1.00	0.96
AAB	A	A	B	228	0.23	0.84	0.86
ABA	A	B	A	101	0.10	0.41	0.67
ABC	A	B	C	155	0.15	0.10	0.65
ABB	A	B	B	105	0.10	0.00	0.65

Table 2: Grouping word occurrences of the test set in 5 configurations. Column Ref shows the correct supersense for a word occurrence, column Lex shows its best lexical scoring supersense and column Cont, its best contextual scoring supersense. Column LM reports the accuracy of the Linear Model (with α equals to 0.7) and column MLP, the accuracy of the MultiLayer Perceptron.

The main missing part in this model is the nature of function Ψ that combines lexical and contextual signatures. We discuss in the two following sections two instantiations of function Ψ.

5.1 Linear Model

The linear model (LM) simply performs a linear combination of vectors $\mathit{LS}(w)$ and $\mathit{CS}(C)$: $\Psi(\mathit{LS}(w), \mathit{CS}(C)) = \alpha \mathit{LS}(w) + (1-\alpha)\mathit{CS}(C)$. This model has one parameter: α, which value has to be estimated on the training corpus. The accuracy on the training set for different values of α has been represented in Table 1. We observe that, when only the lexical score is taken into account ($\alpha = 1$), the model achieves an accuracy of 64.3%. It is equal to 51.83% when the decision is based on the only account of the context score. The optimal value is $\alpha = 0.7$, that reaches an accuracy of 66.35% on the training set and an accuracy of 65% on the test set.

In order to get a better understanding of the results obtained by the linear model, we have grouped the noun occurrences into 5 configurations, described in Table 2 and calculated the accuracy of the linear model for each of them. The configurations compare for each noun occurrence, the correct supersense (column Ref), the best lexical scoring supersense (column Lex) and the best contextual scoring supersense (column Cont). In configuration AAA, all three candidates are equal (Ref=Lex=Cont). In configuration AAB, Lex is correct and Cont is wrong while in configuration ABA, Lex is wrong and Cont is correct. In configuration ABC both Lex and Cont are wrong but they are different, while in configuration ABB they are both wrong and equal to each other. Column 5 reports the number of occurrences that fall in each category. Column 6 gives the ratio of each configuration and column 7 shows the accuracy of LM for every configuration.

The table reveals that in 25% of the cases (configurations ABC and ABB), both L and C are wrong and the linear model behaves very poorly. This was expected, since the model just makes a linear combination of the lexical and contextual scores. The model also behaves poorly in configuration ABA, where Cont should be selected. This is due to the high value of α that tends to favour lexical scores over contextual ones. Linearly combining lexical and contextual signatures with a fixed weight is clearly not an adequate model.

5.2 Multilayer Perceptron

In the MLP model, Ψ is a complex non linear function learned by a neural network that combines the 12 scores that constitute the lexical and contextual signatures. The model chosen is a simple MLP with two hidden layers. Its parameters are learned on the training part of FrSemCor by minimising the categorical cross entropy between the six supersenses. The MLP model achieves an accuracy of 83.02% on the test corpus, an increase of 18.02% absolute with respect to the linear model. The behaviour of the MLP model in the 5 configurations is indicated in the last column of Table 2. As one can see, the predictions

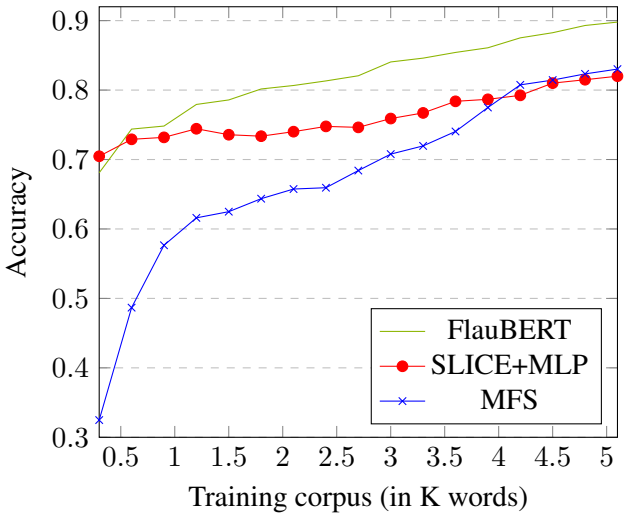

Figure 1: Learning curves for FlauBERT (green), SLICE with the MLP (red), and MFS (blue).

Supersense	Rec.	Prec.	F
ANI	94.02	95.16	94.59
NAT	72.73	86.49	79.01
MAN	75.00	83.61	79.07
INF	79.07	76.84	77.94
DYN	88.82	81.84	85.19
STA	60.87	69.42	64.86
Macro-avg.	78.42	82.23	80.11

	ANI	NAT	MAN	INF	DYN	STA	Σ
ANI	**236**	1	2	2	7	3	251
NAT	2	**32**	1	0	3	6	44
MAN	2	2	**51**	5	7	1	68
INF	3	0	2	**136**	17	14	172
DYN	2	2	3	18	**302**	13	340
STA	3	0	2	16	33	**84**	138
Σ	248	37	61	177	369	121	1013

Table 3: On the left, MLP's precision (Prec.), recall (Rec.) and F-measure (F) per supersense. On the right, supersenses confusion matrix, where reference supersenses index rows and predicted ones index columns.

made in configurations ABC and ABB are much more satisfactory. Accuracy jumps from 10% to 65% in configuration ABC and from 0% to 65% in configuration ABB.

Figure 1 shows the learning curve of SLICE+MLP, the most frequent supersense baseline (MFS), and FlauBERT models. With 300 words in the training set, the MLP model reaches an accuracy of 70%, while the MFS model reaches 31.5%. The difference between the two models decreases as the size of the training set increases. The MFS model accuracy exceeds the MLP's when the size of the training data reaches approximately 4,000 words. FlauBERT is the best performing method after 600 words in the training set, reaching a maximum accuracy of 89.8% on the full training corpus. Notice, however, that FlauBERT embeddings are 85 times larger than ours and were trained on a corpus about 6 times larger than ours. Moreover, the analyses presented in Section 5.3 are only possible in our model, thanks to its interpretability.

Table 3 gives a more detailed view on the MLP predictions. The table on the left hand side displays the precision, recall and F-measure for each supersense. It shows that the model behaves very differently on the different supersenses: supersense ANI obtains the best result, with an F-score of 94.59%, while STA behaves poorly and reaches an F-score of 64.86%, mainly due to its low precision. The confusion matrix on the right hand side reveals that STA is mostly confused with DYN, a confusion partly due to nouns pertaining to both categories, such as *déshydratation* 'dehydration', *reconnaissance* 'recognition/gratitude' or *grossesse* 'pregnancy'.

	Contextual signature						Lexical signature						Ref	MLP
	ANI	NAT	MAN	INF	DYN	STA	ANI	NAT	MAN	INF	DYN	STA		
organisme	0.22	**0.87**	0.19	0.15	0.17	0.09	**0.92**	0.04	0.04	0.29	0.05	0.01	NAT	ANI
demande	0.16	0.04	0.04	**0.46**	0.11	0.04	0.02	0.01	0.07	**0.91**	0.60	0.44	DYN	INF
verger	0.19	**0.90**	0.16	0.01	0.58	0.08	0.29	**0.87**	0.61	0.02	0.13	0.02	MAN	NAT
cas (1)	0.01	0.84	**0.87**	0.26	0.12	0.42	0.09	0.08	0.41	**0.68**	0.23	0.27	STA	MAN
cas (2)	0.58	0.11	0.34	**0.74**	0.57	0.03	0.09	0.08	0.41	**0.68**	0.23	0.27	DYN	INF

Table 4: Contextual and lexical signatures for noun tokens in the test set, where the reference class (4th column) is not the one predicted by the MLP model (5th column).

5.3 Error Analysis

A manual analysis of the results revealed that one key source of errors are nouns having multiple meanings, be they polysemous or multi-facet nouns[11]. They account for 43.4% of the lemmas involved in errors. As a reminder, polysemous nouns have distinct and mutually exclusive meanings. For example, *organisme* 'body/organisation' can denote a natural object (NAT) or an institution (ANI) but a single occurrence of this noun cannot denote both. Multi-facet nouns, on the other hand, have multiple but compatible meanings, a property that can be highlighted by copredication (Cruse, 2002; Ježek and Melloni, 2011). For instance, *demande* 'request' denotes both the request (DYN) and the subject of the request (INF). In some contexts, both facets are triggered, such as *La demande effectuée par la présidente n'a pas été acceptée.* 'The request from the president was not granted'.

Table 4 shows five cases of errors involving multiple meaning words and details their lexical and contextual scores[12]. In row 1, an occurrence of the polysemous noun *organisme* 'body/organisation' is incorrectly labelled ANI instead of NAT. An analysis of the scores reveals that although the context gives a clear preference for the correct sense (NAT), its lexical score is extremely low, while the score of sense ANI is high, provoking the selection of the incorrect sense.

The lexical signature of the word *demande* 'request', in row 2, clearly reflects its multi facet nature, both facets (INF and DYN) obtain the best and second best scores. However, contrary to annotators, the model considered the context as more representative of the INF class.

Another source of errors concerns questionable annotations in the gold data, where decisions made on class delimitation can be debated. For instance, *verger* 'orchard' or *potager* 'vegetable garden' refer to natural objects, but because they are also human creations, they have been classified as MAN in the reference. Our model, however, votes for the NAT class, relying both on contextual and lexical cues. It is interesting to note that the human-made aspect of this natural object seems to be captured in the lexical signature (second best score for MAN).

Gold annotation can also be questioned for nouns that are hard to classify. Notable among those are general nouns such as *fait* 'fact' or *cas* 'case', that can be used to characterise multiple referents and do not clearly pertain to one of the considered supersenses. The two occurrences of *cas*, rows 4 and 5, illustrate these noun properties: the best lexical score is rather low (0.68 for INF) and the gold supersenses, determined by the reference-driven annotation method, are not clearly captured in the contextual signatures.

Linguistic phenomena responsible for the association of several meanings to a single lexical form are thus numerous: homonymy, polysemy, facets, general units having heterogeneous referents. Our interpretable embeddings allow to observe these phenomena and investigate whether these different types of ambiguity or indeterminacy appear as structural properties of our embeddings. They also allow us to take a critical look at the linguistic data we used to learn them, namely the composition of seed lists with respect to the target semantic class, the corpus used to learn lexical signatures or the method used to compute lexical scores. For example, knowing that our model does not classify *organisme* as NAT,

[11] In a word supersense disambiguation task, which consists in selecting the appropriate supersense of a word in a list of predefined supersenses (generally from the Princeton Wordnet), errors are always due to words having multiple meanings. As for supersense tagging, monosemous words can be misclassified, as well as polysemous ones, since word supersenses are not predefined.

[12] Complete sentences are given in Appendix B.

surely because the word is not detected as pertaining to this class in the lexicon, leads us to the following assumptions: nouns related to the body domain may not be well represented in the seed list for NAT; or the body meaning of *organisme* is not frequent enough in frWac, at least in contexts discriminant for the NAT class; or the method we used to compute lexical scores does not properly take into account differences between balanced vs. biased meanings for a given noun. In other words, our model of lexical representation opens the way to several linguistic studies that could allow the prioritisation of ambiguities (e.g., a confusion between the meaning of a polysemous word is more problematic than a confusion between facets of a multi-facet word), and hopefully help supersense tagging and WSD.

6 Conclusions

We have presented a method to learn interpretable embeddings using as weak supervision a list of seed nouns for each supersense. We use the occurrences of seed (prototypical) nouns to train a classifier which associates contexts to supersenses. The context scores are aggregated to generate a single lexical score per supersense. Each of these scores are seen as an interpretable dimension of a dense word embedding.

We have evaluated our method on a supersense tagging task to predict in-context coarse supersenses. In addition to a good performance with very little training data, our method's interpretability allows us to analyse the results in terms of the (supersense) dimensions of the input embeddings. Moreover, our model is considerably faster and lighter than state-of-the-art contextualised embeddings, e.g., we represent inputs as a set of 12 scores whereas FlauBERT uses 1024-dimensional opaque vectors.

We have also built and released the lexicon containing the 10K most frequent French nouns of the frWaC and their corresponding embeddings. We hope that this resource can be complementary to existing embeddings and lexical semantic resources. The lexicon, along with the seed lists, predictions and evaluation data are freely available.[13]

We have applied our method to nouns only, and our embeddings are 6-dimensional (one per coarse supersense), certainly lacking expressive power to cover the full range of semantic distinctions. In theory, there is nothing that prevents us from increasing the number of dimensions (e.g., to cover the traditional Wordnet supersenses), and to experiment with other parts of speech (e.g., verbs and adjectives). In practice, the list of seeds for some supersenses may be too small (e.g., TIME), and we lack annotated corpora to evaluate the method for other POS in French. The sensitivity of the method to the number of seed elements needs to be studied in more detail in the future. Another issue that remains open is the integration of embeddings with different POS tags: should we build a different model per POS (with different interpretable dimensions) or one single models in which inapplicable dimensions are empty?

As future extensions, we envisage integrating our embeddings in other downstream tasks such as semantic parsing. We would like to generalise our method to other syntactic and semantic categories, e.g., can we build interpretable embeddings in which each dimensions represents a given POS using seed lists of verbs, nouns, etc.? Transformer models are a promising alternative to recurrent neural networks to focus on relevant contexts for the classifiers.

Acknowledgements

This work was funded by the French PARSEME-FR project (ANR-14-CERA-0001).[14] We thank Aline Villavicencio and Marco Idiart for their suggestions, as well as the anonymous reviewers.

References

Eneko Agirre, Kepa Bengoetxea, Koldo Gojenola, and Joakim Nivre. 2011. Improving dependency parsing with semantic classes. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 699–703, Portland, Oregon, USA, June. Association for Computational Linguistics.

[13] https://pageperso.lis-lab.fr/carlos.ramisch/?page=downloads/slice

[14] http://parsemefr.lis-lab.fr/

M. Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

L. Barque, P. Haas, R. Huyghe, D. Tribout, M. Candito, B. Crabbé, and V. Segonne. 2020. Annotating a french corpus with supersenses. In *Proceedings of LREC-2020*.

Chris Biemann and Martin Riedl. 2013. Text: now in 2d! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1:55, 07.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, December.

G. Boleda, S. Padó, and J. Utt. 2012. Regular polysemy: A distributional model. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 151–160.

José Camacho-Collados and Mohammad Taher Pilehvar. 2018. From word to sense embeddings: A survey on vector representations of meaning. *J. Artif. Intell. Res.*, 63:743–788.

José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artificial Intelligence*, 240:36–64.

M. Candito and D. Seddah. 2012. Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Proceedings of TALN 2012*, juin.

M. Ciaramita and M. Johnson. 2003. Supersense tagging of unknown nouns in wordnet. In *Proceedings of the EMNLP*, pages 168–175.

D. A. Cruse, 2002. *Aspects of the micro-structure of word meaning*, pages 30–51. Oxford:Oxford University Press.

Claudio Delli Bovi, Jose Camacho-Collados, Alessandro Raganato, and Roberto Navigli. 2017. EuroSense: Automatic harvesting of multilingual sense annotations from parallel text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 594–600, Vancouver, Canada, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado, May–June. Association for Computational Linguistics.

Lucie Flekova and Iryna Gurevych. 2015. Personality profiling of fictional characters using sense-level links between lexical resources. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1805–1816, Lisbon, Portugal, September. Association for Computational Linguistics.

Lucie Flekova and Iryna Gurevych. 2016. Supersense embeddings: A unified model for supersense interpretation, prediction, and utilization. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2029–2041, Berlin, Germany, August. Association for Computational Linguistics.

Z. S. Harris. 1954. Distributional structure. *Word*, 10:146–162.

L. Jahan, G. Chauhan, and M. Finlayson. 2018. A new approach to animacy detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1–12.

Sujay Kumar Jauhar and Eduard Hovy. 2017. Embedded semantic lexicon induction with joint global and local optimization. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 209–219, Vancouver, Canada, August. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pages 3651–3657, Florence, Italy. ACL.

E. Ježek and C. Melloni. 2011. Nominals, polysemy and co-predication. *Journal of cognitive science*, 12:1–31.

Thomas K. Landauer and Susan T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240.

S. Landes, C. Leacock, and R. I. Tengi, 1998. *Building semantic concordances*, pages 1999–2016. The MIT Press.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France, May. European Language Resources Association.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics*.

R. Mihalcea. 2003. The role of non-ambiguous words in natural language disambiguation. In *Proceedings of the Fourth RANLP*, pages 357–366.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. 2013. Efficient estimation of word representations in vector space. In *ICLR Workshop Papers*.

G. Miller, R. Beckwith, C. Fellbaum, Gross D., and K. Miller. 1990. Wordnet: An online lexical database. *International Journal of Lexicography*, 4(3):235–244.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.

R. Navigli and S. P. Ponzetto. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

R. Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):1–69.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1059–1069, Doha, Qatar, October. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), may.

L. Øvrelid. 2006. Towards robust animacy classification using morphosyntactic distributional features. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 47–54.

Alexander Panchenko, Eugen Ruppert, Stefano Faralli, Simone Paolo Ponzetto, and Chris Biemann. 2017. Unsupervised does not mean uninterpretable: The case for word sense induction and disambiguation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 86–98, Valencia, Spain, April. Association for Computational Linguistics.

Tommaso Pasini and Jose Camacho-Collados. 2020. A short survey on sense-annotated corpora. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5759–5765, Marseille, France, May. European Language Resources Association.

J. Pennington, R. Socher, and C. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, LA, USA. Association for Computational Linguistics.

Ashequl Qadir and Ellen Riloff. 2012. Ensemble-based semantic lexicon induction for semantic tagging. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 199–208, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language under-standing by generative pre-training.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works.

Sascha Rothe and Hinrich Schütze. 2015. AutoExtend: Extending word embeddings to embeddings for synsets and lexemes. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1793–1803, Beijing, China, July. Association for Computational Linguistics.

Benoît Sagot and Darja Fišer. 2008. Building a free French wordnet from multilingual resources. In *OntoLex*, Marrakech, Morocco, May.

N. Schneider, D. Hovy, A. Johannsen, and M. Carpuat. 2016. Semeval-2016 task 10: Detecting minimal semantic units and their meanings (dimsum). In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 546–559.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. ACL.

Michael Thelen and Ellen Riloff. 2002. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 214–221. Association for Computational Linguistics, July.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394, Uppsala, Sweden, July. Association for Computational Linguistics.

Dmitry Ustalov, Alexander Panchenko, Chris Biemann, and Simone Paolo Ponzetto. 2019. Watset: Local-global graph clustering with applications in sense and frame induction. *Computational Linguistics*, 45(3):423–479.

David Yarowsky. 1993. One sense per collocation. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

A Supersense Correspondences

Supersense	Wordnet supersenses
Animate entity (ANI)	Animal, Person
Manufactured object (MAN)	Artefact
Natural object (NAT)	Body, Plant, Object
Informational object (INF)	Cognition, Communication, Possession
Dynamic situation (DYN)	Act, Event
Stative situation (STA)	Attribute, State, Feeling

Table 5: List of supersenses proposed in our work, and corresponding Wordnet supersenses.

B Analysed Sentences

- *organisme* (`emea-fr-dev_00335`)

  *En tant que peptide, la bivalirudine est logiquement catabolisée en ses acides aminés constitutifs, avec recyclage ultérieur des acides aminés dans le pool de l'***organisme**.

  'As a peptide, bivalirudin is logically catabolized into its constituent amino acids, with subsequent recycling of the amino acids into the body's pool.'

- *demande* (`frwiki_50.1000_00458`)

  Le 16 juin 2006, les juges Renaud van Ruymbeke et Xavière Simeoni (qui remplace Dominique de Talancé) ont adressé une nouvelle **demande** *de levée du secret défense au ministère de l'Économie et des Finances, seul habilité à saisir la commission consultative du secret de la défense nationale (CCSDN).*

  'On 16 June 2006, Judges Renaud van Ruymbeke and Xavière Simeoni (replacing Dominique de Talancé) sent a new request for the lifting of defence secrecy to the Ministry of Economy and Finance, which is the only body authorised to refer the matter to the Consultative Commission on National Defence Secrecy (CCSDN).'

- *verger* (`annodis.er_00416`)

  La visite du jardin de J.-P. Bruneau est également un moment attendu, lors des portes ouvertes organisées en juin dans ses 1.100 m2 de **verger** *et potager : "Cela permet d'échanger, transmettre un savoir et partager une passion".*

  'The visit of J.-P. Bruneau's garden is also an awaited moment, during the open house days organised in June in its 1.100 m2 of orchard and vegetable garden: "It allows to exchange, to transmit a knowledge and to share a passion".'

- cas(1) (`emea-fr-test_00454`)

  Dans le **cas** *où vous avez eu récemment une fracture de hanche, il est recommandé qu'Aclasta soit administré 2 semaines ou plus après réparation de votre fracture.*

  'If you have had a recent hip fracture, it is recommended that Aclasta be given 2 weeks or more after your fracture is repaired.'

- cas(2)(`emea-fr-test_00176`)

  Aucun **cas** *d'hypocalcémie symptomatique n'a été observé.*

  'No cases of symptomatic hypocalcemia have been observed.'