Evaluation of distributional semantic models: a holistic approach

Gabriel Bernier-Colborne **Patrick Drouin**
Observatoire de linguistique Sens-Texte (OLST), Université de Montréal
C.P. 6128, succ. Centre-Ville, Montréal (QC) Canada, H3C 3J7
{gabriel.bernier-colborne|patrick.drouin}@umontreal.ca

Abstract

We investigate how both model-related factors and application-related factors affect the accuracy of distributional semantic models (DSMs) in the context of specialized lexicography, and how these factors interact. This holistic approach to the evaluation of DSMs provides valuable guidelines for the use of these models and insight into the kind of semantic information they capture.

1 Introduction

Distributional semantic models (DSMs) can be very useful tools for specialized lexicography, as they can help identify semantic or conceptual relations between terms based on corpus data, among other uses. The quality of the results produced by these models depends on two types of factors: model-related factors and application-related factors. First, they depend on the type of model and the settings used for each of the model's (hyper)parameters. Second, they depend on various aspects of the target application. In the case of specialized lexicography, these factors include the kinds of terms that will be included in the lexical resource and the kinds of relations that will be described therein. The target relations can include typical paradigmatic relations such as (near-)synonymy (e.g. *preserve→protect*), but also others such as syntactic derivation (e.g. *preserve→preservation*). There may also be interactions between the various factors: for instance, the optimal parameter settings may depend on the target relations.

We investigated how these two types of factors affect the quality of the results produced by DSMs, and how they interact, i.e. how various aspects of specialized lexicography must be accounted for when choosing and tuning a model. The aspects considered in this paper are the the part-of-speech (POS) of the terms included in the resource, the descriptive framework, and the target relations. To this end, we carried out an experiment in which DSMs were built on domain-specific corpora and evaluated on gold standard data we extracted from specialized dictionaries.

2 Related work

Numerous studies have addressed the evaluation and optimization of DSMs. These studies tend to focus on model-related factors, by comparing different models or analyzing the influence of their (hyper)parameters, although some studies use several different tasks or datasets for evaluation purposes (Bullinaria and Levy, 2007; Bullinaria and Levy, 2012; Kiela and Clark, 2014; Baroni et al., 2014), thereby taking the target application into account to some extent. Studies that systematically assess the influence of both model-related and application-related factors are relatively rare. In the case of the DSM which we refer to as the bag-of-words (BOW) model, research conducted as early as the 1960s showed that its parameters, such as the size of the context window, affected the kinds of semantic relations that were captured (Moskowich and Caplan, 1978). Systematic evaluations of DSMs have recently been carried out, some of which take into account the target relations (Sahlgren, 2006; Lapesa et al., 2014) or the POS (Hill et al., 2014; Tanguy et al., 2015). These studies tend to show that the accuracy of DSMs depends on such application-related factors, as do their optimal (hyper)parameter settings. Our work is

Proceedings of the 5th International Workshop on Computational Terminology,
pages 52–61, Osaka, Japan, December 12 2016.

related to these studies, but takes into account a wider range application-related factors, including the descriptive framework, and systematically evaluates how they affect accuracy and how they interact with model-related factors. Furthermore, the target relations considered include not only typical paradigmatic relations, but also syntactic derivation (see Section 3). This relation has not been studied in the context of DSM evaluation as far as we know, and it is not represented in the datasets that are commonly used to evaluate DSMs. The analogy dataset used by Mikolov et al. (2013a) does include adjective-adverb morphological derivatives, but we do not know of any commonly used datasets that cover morphological derivation more extensively, nor any that represent syntactic derivation specifically.

This study contains a comparative evaluation of two differents DSMs, namely the BOW model and the neural word embeddings produced by `word2vec` (Mikolov et al., 2013a; Mikolov et al., 2013b). Several such evaluations have been carried out recently. Baroni et al. (2014) compared the BOW model and `word2vec`[1] on several datasets and found that `word2vec` systematically provided better results. However, the word representations they made available were evaluated by Ferret (2015) on a different dataset, and the BOW model performed better. Levy et al. (2015) showed that when the models' (hyper)parameters are tuned correctly, the BOW model and `word2vec`[2] provide similar accuracy, and the best model depends on the task used for evaluation purposes. To our knowledge, the ability of these two types of DSM to detect various semantic relations has not been evaluated systematically. This is one of the contributions of this study. Moreover, we investigate how various application-related factors come into play when tuning `word2vec`'s hyperparameters. Another original aspect of this work is that we compare the two DSMs on domain-specific data.

3 Data

The corpus used to build the models is a specialized corpus on the environment which is freely available to researchers, called the PANACEA Environment English monolingual corpus[3] (ELRA-W0063). The corpus was compiled automatically using a focused web crawler (Prokopidis et al., 2012). Basic preprocessing was applied, which included extracting the text from the XML files that comprise the corpus[4], replacing non-ASCII characters with ASCII equivalents[5], lemmatizing[6] and converting to lower case.

Models were evaluated using two types of evaluation data[7] (or gold standards), that represent two descriptive frameworks, namely a lexico-semantic approach to terminology (L'Homme, 2004) and frame semantics (Fillmore, 1982). These datasets, which were extracted from specialized dictionaries on the environment domain, are comprised of pairs of semantically related terms or sets of terms that evoke the same semantic frame (e.g. the frame Change_of_temperature is evoked by terms such as *cool, cooling, warm,* and *warming*.) respectively. We created 7 different datasets of semantic relations and one dataset for frame-evoking terms. These datasets, which are described in Table 1, are comprised of query terms mapped to a set of related terms. Models are evaluated by computing the nearest neighbours of each query and looking up the query's related terms in this sorted list of neighbours.

The semantic relations were extracted from DiCoEnviro[8]. We extracted four kinds of semantic relations, namely (near-)synonyms, antonyms, hypernyms/hyponyms and syntactic derivatives. The first three types of relations are typical paradigmatic relations that involve two terms of the same POS. Syntactic derivatives (Mel'čuk et al., 1995, p. 133) are terms that have the same meaning, but belong to different POS, and thus have different syntactic behaviours – they may be morphologically related, but this need not be the case (e.g. *city* and *urban*). A dataset was created for each of these four relations. We also created three datasets for the three POS we took into account, namely nouns, verbs, and adjectives.

[1] More specifically, the CBOW architecture.

[2] Here, the skip-gram architecture was used rather than the CBOW architecture.

[3] `http://catalog.elra.info/product_info.php?products_id=1184`

[4] Documents containing less than 50 words were excluded.

[5] We use the Unidecode Python library (`https://pypi.python.org/pypi/Unidecode`).

[6] TreeTagger (Schmid, 1994) was used for lemmatization.

[7] We have made these datasets available, as well as the code we developed for this study. See `https://github.com/gbcolborne/exp_phd`.

[8] `http://olst.ling.umontreal.ca/cgi-bin/dicoenviro/search-enviro.cgi?ui=en`

These contain all the relations between two terms of a given POS (so they do not contain any syntactic derivatives). As for the sets of frame-evoking terms, these were extracted from the Framed DiCoEnviro[9].

Name	Queries	Relations	Description
QSYN	282	517	Synonyms, near-synonyms, co-hyponyms or term variations, e.g. *green*: {*alternative, clean, pure, smart*}.
ANTI	77	109	Antonyms, e.g. *absorb*: {*emit, radiate, reflect*}.
HYP	61	87	Hyponyms and hypernyms, e.g. *precipitation*: {*rain, snow, hail*}.
DRV	174	175	Syntactic derivatives, e.g. *adaptive*: {*adapt, adaptation*}.
NN	190	404	Nouns are mapped to all related nouns (QSYN, ANTI or HYP).
VV	84	187	Verbs are mapped to all related verbs (QSYN or ANTI).
JJ	67	122	Adjectives are mapped to all related adjectives (QSYN or ANTI).
SETS	168	480	Frame-evoking terms are mapped to terms that evoke the same frame, e.g. *warming*: {*warm, cool, cooling*}.

Table 1: Datasets used for evaluation.

It is important to note that only single-word terms were included in these datsets. For various reasons, we decided not to include any multi-word terms in the target words that were evaluated (see Section 4), and only terms that were among these target words were included in the gold standard datasets. Multi-word terms could be included among the target words if required by the target application. Compositionality-based methods (Mitchell and Lapata, 2008; Baroni and Zamparelli, 2010; Mikolov et al., 2013b; Weeds et al., 2014) could also be used to account for multi-word terms.

4 Methodology

The experiment we carried out involves a comparative evaluation of two DSMs and a systematic exploration of their (hyper)parameters. Both of these models produce vector representations of words based on the contexts in which they appear in a corpus, the underlying hypothesis being that words that appear in similar contexts have similar meanings (Harris, 1954; Firth, 1957). Words that appear in similar contexts will thus have similar vector representations, and the semantic similarity of any two words can then be estimated by computing the similarity of their vectors.

The contexts of a word can be defined in various ways. In both of the DSMs we evaluated, the contexts of a word are the words that co-occur with it. Since the contexts are also words, we will sometimes call them *context words*. In this work, we use a sliding context window to determine which words co-occur. The context window spans a certain number of words on either side of a given word token.

The first DSM we evaluated is a simple vector space model which has been studied extensively in the past few decades (Schütze, 1992; Lund et al., 1995; Sahlgren, 2006; Lapesa et al., 2014, inter alia), but whose origins can be traced back to the 1960s (Harper, 1965; Moskowich and Caplan, 1978). We will call this the bag-of-words (BOW) model. To build a BOW model, we compute a matrix M in which value M_{ij} is the weighted cooccurrence frequency of word w_i and context c_j. Various weighting schemes can be used, one popular choice being positive pointwise mutual information (PPMI). Each word w_i is represented by a vector $M_{i:}$ in which each value represents the association strength of w_i and a specific context word. The matrix M can be transformed in other ways once the cooccurrence frequencies have been counted and weighted, e.g. by applying some form of dimensionality reduction, but in this work, we use the basic BOW model, in which words are represented by sparse, high-dimensional vectors.

The second DSM we evaluated is built using the neural probabilistic language model known as `word2vec`. This model learns distributed word representations (often called *embeddings*) which can be used in the same way as BOW vectors to estimate the semantic similarity of words. These representations are learned by training a neural network that aims to predict each word token based on its contexts

[9] `http://olst.ling.umontreal.ca/dicoenviro/framed/index.php`

(co-occurring words). An alternative approach aims to predict the contexts of each word token. These two architectures are known as *continuous bag-of-words* (CBOW) and skip-gram respectively.

As with all DSMs, the BOW model and `word2vec` have several (hyper)parameters that must be set in order to build or train a model. We have already mentioned three such parameters: the size of the context window, the weighting scheme (for the BOW model), and the architecture (for `word2vec`). These parameters have an effect on the word representations that are produced, and on the accuracy of the word similarity scores we obtain by comparing the word representations.

In order to assess the influence of the (hyper)parameters of both DSMs, we tried several settings for each parameter and evaluated every possible combination of these parameter settings.

For the BOW model, we examined three parameters related to the context window. The context window has not only a size, but a shape, which is a function that determines the increment that is added to the cooccurrence frequency of a given (word, context) pair, based on the distance between word and context. In a rectangular window, this increment is always 1, regardless of distance. In a triangular window, the increment is inversely proportional to the distance between the word and the context: 1 if the distance is 1 word, $\frac{1}{2}$ if the distance is 2, and so forth. The window also has a direction: we can look left, right, or in both directions. In the latter case, we can sum the frequencies observed left and right of a given word, or encode these frequencies separately, in which case the matrix M contains two dimensions for each context word, one for each direction. These two types of windows are sometimes called left+right (L+R) and left&right (L&R).

We also assessed the influence of the weighting scheme. This is usually an association measure such as mutual information. We tested the 6 simple association measures defined in Evert's (2007, ch. 4) work on collocations. These measures compare the observed cooccurrence frequency (O) of two words to their expected cooccurrence frequency (E). For instance, (pointwise) mutual information is defined as $\text{MI} = \log_2 \left(\frac{O}{E}\right)$. If O is much greater than E, this suggests a strong association between the two words. We use Evert's definitions for all these measures, but calculate E somewhat differently:

$$E(w_i, c_j) = \frac{\sum_{j'} M_{ij'} \sum_{i'} M_{i'j}}{\sum_{i'} \sum_{j'} M_{i'j'}}$$

where M is the unweighted cooccurrence frequency matrix. Negative association scores were always set to 0 (so MI becomes PPMI). A transformation (log or sqrt, where $\log(x) = \ln(x+1)$ and $\text{sqrt}(x) = \sqrt{x}$) was applied to some of the association measures, following Lapesa et al. (2014), and based on our own preliminary experiments. We also tried applying a simple log transformation to the cooccurrence frequencies, without applying an association measure beforehand.

The settings we tested for each of the four parameters are:

- Type of context window: L+R or L&R.
- Size of context window: 1-10 words.
- Shape of context window: rectangular or triangular.
- Weighting scheme: log, MI, MI^2, MI^3, log(local-MI), log(simple-LL), sqrt(t-score), sqrt(z-score).

In the case of `word2vec`, we examined the five hyperparameters that have an important effect on performance according to the documentation of `word2vec`[10]. The architecture used to learn the word embeddings is one of these hyperparameters. We must also select a training algorithm: whatever the architecture, the model can be trained using a hierarchical softmax function, or by sampling negative examples (or classes), in which case we also have to choose the number of negative samples. `word2vec` also provides a function that subsamples frequent words, i.e. words whose relative frequency in the corpus is greater than some threshold. This function randomly deletes occurrences of these frequent words before the model is trained, each occurrence having a certain probability of being deleted, which depends on the word's frequency. The last two hyperparameters are the dimensionality of the word embeddings and the size of the context window. The settings we tested for each hyperparameter are:

[10] `https://code.google.com/p/word2vec/`

- Architecture: CBOW or skip-gram.
- Negative samples: 5, 10 or none (hierarchical softmax is used instead).
- Subsampling threshold: low (10^{-5}), high (10^{-3}) or none (no subsampling).
- Size of context window: 1-10 words.
- Dimensionality of word embeddings: 100 or 300.

A few more details regarding the training and evaluation of the two DSMs may be worth mentioning. In the case of the BOW model, the set of context words contained all the target words that were used for evaluation purposes. These target words (for both models) were the 10K most frequent words in the (lemmatized) corpus, excluding stop words and words that contained any character other than a letter, a digit or a hyphen. In the case of the BOW model, out-of-vocabulary words were not deleted, simply ignored, and the context window was allowed to span sentence boundaries. For `word2vec`, we used the `word2vec` software as is, using the default settings for all hyperparameters except those whose influence we investigated. It is also worth noting that the context window implemented in `word2vec` has a shape that gives more weight to contexts that are closer to a given word (similar to a triangular window) – this is implemented by drawing the effective window size for a given token uniformly between 1 and the size specified by the user (Levy et al., 2015).

The measure we used to evaluate the models is mean average precision[11] (MAP). This measure tells us how accurate the sorted list of neighbours we get for a given query is, based on the rank of its related terms according to the gold standard. The nearer the related terms are to the top of this list on average for each of the queries, the higher the MAP. The sorted list of neighbours is obtained by computing the similarity (or distance) between the query's vector and the vectors of all other target words. We use the cosine similarity (Salton and Lesk, 1968), which is the most commonly used measure for distributional similarity (Turney and Pantel, 2010). The sorted list of neighbours is then evaluated on the various datasets.

5 Results

First, we compare the BOW model and `word2vec` (W2V), by observing the MAP of each model on each of the datasets. The maximum MAP achieved by each model is shown in Table 2. These results show that the BOW model achieves a higher MAP than W2V on the three paradigmatic relations (QSYN, ANTI, and HYP) if its parameters are tuned correctly, but W2V achieves a much higher MAP on DRVs. In other words, the BOW model is better at estimating the semantic similarity of terms that have similar syntactic behaviours, whereas W2V is better at estimating the similarity of terms that have different syntactic behaviours, but the same meaning[12]. Furthermore, the BOW model produces a higher MAP than W2V on all three parts-of-speech (when only paradigmatic relations are considered) on average, though the best W2V model on verbs has a slightly higher MAP than the best BOW model. As for the sets of frame-evoking terms (SETS), W2V achieves a higher accuracy, but the BOW model performs slightly better on average.

Dataset	BOW	W2V
QSYN	**0.418** (0.321 $\pm$ 0.056)	0.396 (0.298 $\pm$ 0.042)
ANTI	**0.383** (0.247 $\pm$ 0.056)	0.321 (0.228 $\pm$ 0.039)
HYP	**0.252** (0.211 $\pm$ 0.017)	0.199 (0.153 $\pm$ 0.019)
DRV	0.458 (0.328 $\pm$ 0.080)	**0.544** (0.347 $\pm$ 0.118)
NN	**0.398** (0.329 $\pm$ 0.045)	0.373 (0.299 $\pm$ 0.034)
VV	0.326 (0.255 $\pm$ 0.048)	**0.329** (0.239 $\pm$ 0.046)
JJ	**0.501** (0.317 $\pm$ 0.086)	0.454 (0.274 $\pm$ 0.050)
SETS	0.326 (0.282 $\pm$ 0.026)	**0.348** (0.275 $\pm$ 0.031)

Table 2: Maximum MAP (with average and std. dev. in brackets) of BOW and W2V models on each dataset.

[11] See `http://goo.gl/qdlQ7n`.

[12] This may be due to the dimensionality reduction that occurs in the `word2vec` model.

If we compare the maximum MAP obtained on each of the datasets (by either BOW or W2V), we see that DSMs capture syntactic derivatives even more accurately than near-synonyms if the models are tuned for this relation. Antonyms are captured almost as accurately as synonyms, but the MAP obtained on hypernyms/hyponyms is quite a bit lower. As for the POS, DSMs model adjectives most accurately, followed by nouns, then verbs. The MAP achieved on the SETS is lower than on all the semantic relations except for hypernyms/hyponyms. This is due to at least two factors. First, the SETS contain a relatively high number of verbs, and as we have seen, verbs are the most challenging POS for these two DSMs. Second, the sets of frame-evoking terms represent a mixture of syntactic derivation and typical paradigmatic relations, especially synonymy, and although we achieve a high MAP on both of these relations, the (hyper)parameter settings that work best for each are very different, as we will show below.

Now that we have assessed the quality of the results with respect to various aspects of the target application (the descriptive framework, the target relations, the POS) and compared the two DSMs, we turn our attention to the influence of their (hyper)parameters. For each such parameter, we will observe the average MAP for each setting of that parameter. We use the average MAP instead of the maximum in order to determine which settings produce consistently good results, regardless of the settings used for the other parameters. Interactions between the parameters are not accounted for in the analysis presented in this paper.

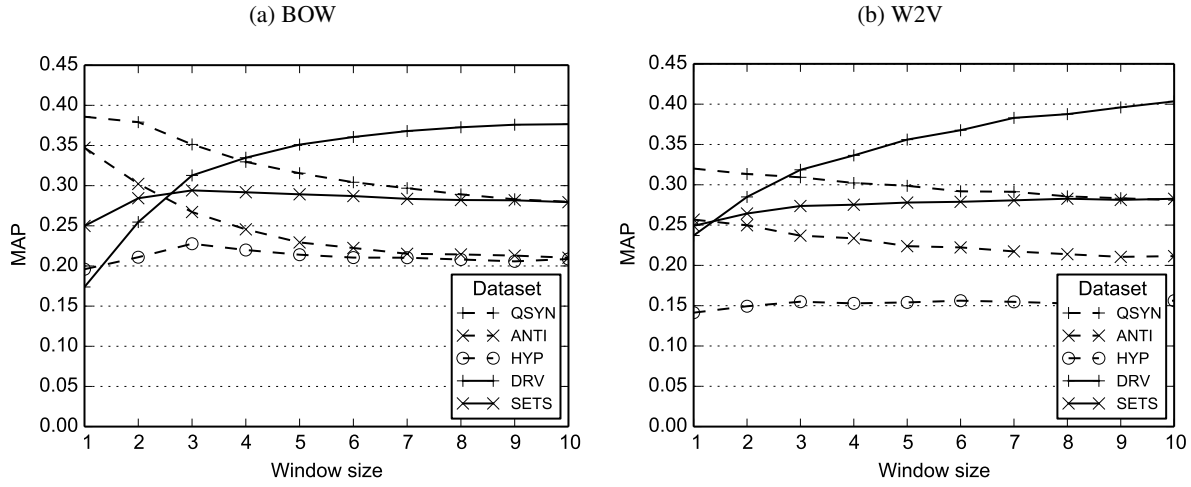

Figure 1: Average MAP of **(a)** BOW and **(b)** W2V models wrt window size.

The influence of the window size on the accuracy of both DSMs is illustrated in Figure 1. This figure shows that for the three paradigmatic relations (QSYN, ANTI, and HYP), the optimal window size is small, i.e. 1-3 words. Though the figure does not show the results for each POS separately, this is true for every POS. The optimal size is 1 for adjectives, and accuracy quickly drops off as window size increases. The optimal size is 1 for verbs also, and 1 or 3 for nouns (BOW and W2V respectively). On the other hand, the optimal window size for DRVs is quite large. The average MAP does not seem to have peaked even with a window size of 10, however the maximum MAP we observed was achieved with a window of 9 words (with both models). Thus, narrow windows capture paradigmatic relations most accurately, but wider windows are better for syntactic derivatives. This may be due to a tendency of syntactic derivatives to co-occur, as wider windows lead to co-occurring words having more similar distributional representations. For instance, if we observe the sequence of words `a x y b`, then `a` is a context of both `x` and `y` (if the window size is at least 2), and so is `b`. Every time `x` and `y` appear next to each other (or close enough, depending on the size of the window), they share contexts, which increases the similarity of their representations.

As for sets of frame-evoking terms, the window size should be at least 3, but the average MAP does not vary much with respect to window size beyond this point. As the window size increases, accuracy improves on DRVs, but worsens on paradigmatic relations, such that accuracy on the SETS, which

represent a mixture of these relations, remains relatively stable.

The figure also shows that the influence of the window size is very similar in the BOW and W2V models. We could investigate whether this is the case for other (hyper)parameters that are applicable to both models (e.g. the window shape) or can be adapted from one model to the other (e.g. context distribution smoothing for the negative sampling function (Levy et al., 2015)). Instead of comparing the influence of the same parameters in both models, we chose to investigate the influence of a set of parameters that are typical of each model. Our observations on the influence of the window size suggest that the influence of parameters that are common to both DSMs would be very similar.

Parameter	Setting	QSYN	ANTI	HYP	DRV	NN	VV	JJ	SETS
Window type	L&R	**0.332**	**0.257**	**0.213**	0.288	**0.336**	**0.266**	**0.334**	0.274
	L+R	0.311	0.237	0.209	**0.368**	0.323	0.244	0.301	**0.291**
Window shape	Rectangular	0.297	0.223	0.209	**0.337**	0.310	0.233	0.282	0.273
	Triangular	**0.346**	**0.270**	**0.213**	0.320	**0.348**	**0.276**	**0.352**	**0.292**
Weighting scheme	None	0.172	0.196	0.070	0.205	0.168	0.160	0.180	0.182
	log	0.266	0.211	0.201	0.304	0.272	0.206	0.277	0.258
	MI	0.321	0.239	0.212	**0.353**	0.338	0.254	0.303	0.283
	MI^2	0.308	0.238	**0.224**	0.292	0.315	0.243	0.309	0.264
	MI^3	0.300	0.232	0.215	0.302	0.304	0.235	0.306	0.271
	log(local-MI)	0.348	0.258	0.210	0.343	0.353	0.277	0.337	0.294
	log(simple-LL)	**0.349**	0.261	0.209	0.347	**0.354**	**0.281**	0.339	**0.301**
	sqrt(t-score)	0.341	0.261	0.220	0.338	0.352	0.272	0.327	0.288
	sqrt(z-score)	0.338	**0.273**	0.198	0.345	0.345	0.270	**0.342**	0.300

Table 3: Average MAP of BOW models wrt to window type, window shape, and weighting scheme.

.

Table 3 shows the influence of the three other parameters of the BOW model: the type of window, its shape, and the weighting scheme. In the latter case, we added the results we would obtain without weighting the cooccurrence frequencies, in order to show the importance of using some kind of weighting scheme, but it is important to note that the unweighted models were not included in the rest of the analysis presented in this paper. Indeed, using some kind of weighting scheme always improves accuracy, even a simple log transformation, though the association measures almost always provide better results. Interestingly, MI (aka PPMI), which is likely the most common weighting scheme in this kind of DSM, is not among the best-performing schemes, except on one dataset: DRVs. MI is known to have a low-frequency bias (Evert, 2007, p. 19), which appears to be beneficial in the case of syntactic derivatives, whereas near-synonyms and antonyms are detected more accurately using measures which do not have this bias, such as simple-LL.

The shape of the window is another parameter whose optimal setting is different for syntactic derivatives than for other semantic relations. Whereas the triangular window works best for QSYNs and ANTIs, on average, DRVs are detected more accurately using a rectangular window. Since DRVs prefer a wider window, as we have already shown, it intuitively makes sense that they would prefer a rectangular window, as it gives more weight to long-distance contexts than a triangular window.

As for the window type, we again observe a difference between DRVs and other semantic relations. Indeed, the L+R works much better than the L&R window for DRVs, whereas the L&R provides better results for QSYNs and ANTIs, on average. We propose the following explanation. A pair of DRVs are likely to have some collocates in common, but these may appear on opposite sides of the two words (e.g. compare *to emit GHGs* and *GHG emissions*). If the cooccurrence frequencies for the left and right contexts are encoded separately, i.e. if we use a L&R window, the model may not adequately represent the fact that these words have similar collocates. This would explain why the L+R window works better for DRVs.

Hyperparameter	Setting	QSYN	ANTI	HYP	DRV	NN	VV	JJ	SETS
Architecture	skip-gram	0.287	0.226	**0.154**	**0.390**	0.293	0.225	0.266	**0.283**
	CBOW	**0.308**	**0.229**	0.152	0.304	**0.304**	**0.253**	**0.283**	0.266
Negative samples	None	0.284	0.227	0.150	0.333	0.284	0.226	0.274	0.266
	5	0.302	0.227	0.154	0.349	0.305	0.244	0.271	0.276
	10	**0.307**	**0.229**	**0.155**	**0.359**	**0.308**	**0.246**	**0.279**	**0.282**
Subsampling threshold	None	**0.323**	**0.258**	0.152	0.251	**0.316**	**0.267**	**0.307**	0.258
	Low	0.254	0.184	0.149	**0.457**	0.267	0.188	0.225	**0.285**
	High	0.316	0.242	**0.157**	0.334	0.313	0.261	0.291	0.282
Dimensionality	100	0.284	0.228	0.145	0.316	0.285	0.229	0.264	0.255
	300	**0.311**	0.228	**0.160**	**0.379**	**0.312**	**0.248**	**0.285**	**0.294**

Table 4: Average MAP of W2V models wrt the architecture, the number of negative samples for training, the threshold for subsampling and the dimensionality of word embeddings.

Thus, the influence of all four parameters that we have examined in the case of the BOW model is different for DRVs than for near-synonyms and other paradigmatic relations. In the case of W2V, three of the five hyperparameters considered in this study also exhibit such a difference. We have already shown that DRVs prefer wide context windows whereas narrow windows capture paradigmatic relations more accurately. Table 4 shows the influence of the four other hyperparameters. Regarding the neural network's architecture, CBOW works best, on average, for QSYNs, but skip-gram works best for DRVs. As for the subsampling function, it provides little or no gains on the three paradigmatic relations[13], but dramatically increases accuracy on DRVs, especially if the frequency threshold is low, which leads to a more "aggressive" subsampling. Inversely, aggressive subsampling results in quite a large drop in accuracy for QSYNs and ANTIs. Finally, the optimal settings for the dimensionality of the word embeddings and for the training algorithm are the same on all datasets: 300-dimensional embeddings perform better than 100-dimensional ones, and negative sampling works better than a hierarchical softmax, the MAP improving slightly if we use 10 samples rather than 5.

6 Concluding remarks

In this paper, we presented the results of a holistic approach to the evaluation of DSMs in the context of specialized lexicography. We investigated how both model-related and application-related factors affect the quality of the results produced by DSMs, and how they interact. By evaluating models on datasets representing different semantic relations, we showed that DSMs capture syntactic derivatives even better than typical paradigmatic relations such as synonymy, but that the model and (hyper)parameter settings that perform best for these two types of relations are very different. Our results also indicate that verbs are more challenging for DSMs than nouns and adjectives. Furthermore, we showed that the quality of the results depends on the descriptive framework used for the lexical resource being developed. Accuracy was lower on sets of frame-evoking terms than on every semantic relation we considered except hypernymy/hyponymy. This is due to at least two reasons. Sets of frame-evoking terms represent a mixture of syntactic derivation and typical paradigmatic relations such as synonymy, and since the best models for these two types of relations are very different, the ability of a single model to capture terms that evoke the same frame is limited. Furthermore, a high percentage of frame-evoking terms are verbs, which are challenging for DSMs.

Although we only presented the results obtained on English data in this paper, we also conducted this experiment on French data, and the results, a part of which we reported in another paper (Bernier-Colborne and Drouin, 2016b), are very similar.

[13]It is worth remembering that we only tested two values for the frequency threshold, these being the limits of the range of recommended values. Other settings might provide better results.

This work provides valuable guidelines for the use of DSMs for lexicographical purposes. It also provides new insights into the kind of semantic information that is captured by these models. Extensions of this work could include testing other DSMs, other (hyper)parameters, or other settings; and evaluating on different tasks or data from different domains. Based on the work presented in this paper, we investigated whether different DSMs could be combined in order to improve accuracy, and showed that combining the best BOW and W2V models increased the MAP on the sets of frame-evoking terms (Bernier-Colborne and Drouin, 2016a).

Acknowledgements

This work was supported by the Social Sciences and Humanities Research Council (SSHRC) of Canada.

References

Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1183–1193. ACL.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 238–247, Baltimore. ACL.

Gabriel Bernier-Colborne and Patrick Drouin. 2016a. Combiner des modèles sémantiques distributionnels pour mieux détecter les termes évoquant le même cadre sémantique [in French]. In *Proceedings the 23rd French Conference on Natural Language Processing (TALN)*, pages 381–388.

Gabriel Bernier-Colborne and Patrick Drouin. 2016b. Évaluation des modèles sémantiques distributionnels : le cas de la dérivation syntaxique [in French]. In *Proceedings the 23rd French Conference on Natural Language Processing (TALN)*, pages 125–138, Paris.

John A. Bullinaria and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior research methods*, 39(3):510–526.

John A. Bullinaria and Joseph P. Levy. 2012. Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD. *Behavior research methods*, 44(3):890–907.

Stefan Evert. 2007. Corpora and collocations (extended manuscript). In Anke Lüdeling and Merja Kytö, editors, *Corpus Linguistics. An International Handbook*, volume 2. Walter de Gruyter, Berlin/New York. `http://www.stefan-evert.de/PUB/Evert2007HSK_extended_manuscript.pdf`.

Olivier Ferret. 2015. Réordonnancer des thésaurus distributionnels en combinant différents critères [in French]. *TAL*, 56(2):21–49.

Charles J. Fillmore. 1982. Frame semantics. In The Linguistic Society of Korea, editor, *Linguistics in the Morning Calm: Selected Papers from SICOL-1981*, pages 111–137. Hanshin Publishing Co., Seoul.

John Rupert Firth. 1957. A synopsis of linguistic theory 1930–1955. In The Philological Society, editor, *Studies in Linguistic Analysis*, pages 1–32. Blackwell, Oxford.

Kenneth E. Harper. 1965. Measurement of similarity between nouns. In *Proceedings of the 1965 Conference on Computational Linguistics (COLING)*, pages 1–23, Bonn. ACL.

Zellig S. Harris. 1954. Distributional structure. *Word*, 10(2–3):146–162.

Felix Hill, Roi Reichart, and Anna Korhonen. 2014. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *CoRR*, abs/1408.3456.

Douwe Kiela and Stephen Clark. 2014. A systematic study of semantic vector space model parameters. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) @ EACL 2014*, pages 21–30. ACL.

Gabriella Lapesa, Stefan Evert, and Sabine Schulte im Walde. 2014. Contrasting syntagmatic and paradigmatic relations: Insights from distributional semantic models. In *Proceedings of the Third Joint Conference on Lexical and Computational Semantics (*SEM 2014)*, pages 160–170, Dublin. ACL/DCU.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Marie-Claude L'Homme. 2004. A lexico-semantic approach to the structuring of terminology. In *Proceedings the 3rd International Workshop on Computational Terminology (CompuTerm)*, pages 7–14.

Kevin Lund, Curt Burgess, and Ruth Ann Atchley. 1995. Semantic and associative priming in high-dimensional semantic space. In *Proceedings of the 17th Annual Conference of the Cognitive Science Society*, pages 660–665.

Igor' Aleksandrovič Mel'čuk, André Clas, and Alain Polguère. 1995. *Introduction à la lexicologie explicative et combinatoire [in French]*. Duculot, Louvain-la-Neuve.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at the International Conference on Learning Representations (ICLR)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26 (NIPS)*, pages 3111–3119. Curran Associates, Inc.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *Proceedings of ACL-08: HLT*, pages 236–244.

Wolf Moskowich and Ruth Caplan. 1978. Distributive-statistical text analysis: A new tool for semantic and stylistic research. In G. Altmann, editor, *Glottometrika*, pages 107–153. Studienverlag Dr. N. Brockmeyer, Bochum.

Prokopis Prokopidis, Vassilis Papavassiliou, Antonio Toral, Marc Poch Riera, Francesca Frontini, Francesco Rubino, and Gregor Thurmair. 2012. Final report on the corpus acquisition & annotation subsystem and its components. Technical Report WP-4.5, PANACEA Project.

Magnus Sahlgren. 2006. *The word-space model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. Ph.D. thesis, Stockholm University.

Gerard Salton and Michael E. Lesk. 1968. Computer evaluation of indexing and text processing. *Journal of the ACM*, 15(1):8–36.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*.

Hinrich Schütze. 1992. Dimensions of meaning. In *Proceedings of the 1992 ACM/IEEE Conference on Supercomputing (Supercomputing'92)*, pages 787–796.

Ludovic Tanguy, Franck Sajous, and Nabil Hathout. 2015. Évaluation sur mesure de modèles distributionnels sur un corpus spécialisé : comparaison des approches par contextes syntaxiques et par fenêtres graphiques [in French]. *TAL*, 56(2):103–127.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Julie Weeds, David Weir, and Jeremy Reffin. 2014. Distributional composition using higher-order dependency vectors. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) @ EACL*, pages 11–20.