One of these spɹoʍ is not like the other: a reproduction of outlier identification using non-contextual word representations

Jesper Brink Andersen*
Aarhus University
jesperbrink@post.au.dk

Mikkel Bak Bertelsen*
Aarhus University
mikkelbak@post.au.dk

Mikkel Hørby Schou*
Aarhus University
mikkelschou@post.au.dk

Manuel R. Ciosici
Information Sciences Institute
IT University of Copenhagen
manuelc@isi.edu

Ira Assent
Aarhus University
ira@cs.au.dk

Abstract

Word embeddings are an active topic in the NLP research community. State-of-the-art neural models achieve high performance on downstream tasks, albeit at the cost of computationally expensive training. Cost aware solutions require cheaper models that still achieve good performance. We present several reproduction studies of intrinsic evaluation tasks that evaluate non-contextual word representations in multiple languages.

Furthermore, we present 50-8-8, a new data set for the outlier identification task, which avoids limitations of the original data set, such as ambiguous words, infrequent words, and multi-word tokens, while increasing the number of test cases. The data set is expanded to contain semantic and syntactic tests and is multilingual (English, German, and Italian).

We provide an in-depth analysis of word embedding models with a range of hyper-parameters. Our analysis shows the suitability of different models and hyper-parameters for different tasks and the greater difficulty of representing German and Italian languages.

1 Introduction

Unsupervised word embeddings have largely replaced language-specific hand-designed representations of syntax and semantics (Mikolov et al., 2013a; Levy and Goldberg, 2014a; Devlin et al., 2019). Models based on deep neural networks such as the BERT family (Devlin et al., 2019; Liu et al., 2019; Sanh et al., 2019) construct contextualized word vector representations. Showing state-of-the-art results in benchmarks such as GLUE (Wang et al., 2018), they are computationally expensive for both training and inference (Devlin et al., 2019; You et al., 2020) with significant cost for the environment (Strubell et al., 2019). In this paper, we turn our attention back to the non-contextual, less resource-hungry word representations of the word2vec family (Mikolov et al., 2013a; Levy and Goldberg, 2014a).

We contribute reproduction studies on the quality of the non-contextual word representations using outlier identification (Camacho-Collados and Navigli, 2016) and the classic word analogy task (Mikolov et al., 2013a). Replicability and reproducibility have gained increasing importance in the NLP community: focus on the publication of code and data with papers, special sections in leading journals (Branco et al., 2017), and dedicated shared tasks (Branco et al., 2020). Unfortunately, there exist opposing definitions of the terms *reproduction* and *replication* (e.g., Branco et al. (2017) and Chris (2009)), while others propose a spectrum of reproducibility (Peng, 2011). While we aim to reproduce the experiments in our target papers closely, we go beyond a straight-forward reproduction and address further questions such as effect of hyper-parameters, linear contexts (CBOW vs. skip-gram), and non-linear dependency-based contexts (word2vecf).

We also propose 50-8-8, an alternative to the 8-8-8 outlier identification data set (Camacho-Collados and Navigli, 2016) that is several times larger, includes both semantic and syntactic evaluations, and addresses result variance issues that affect the original 8-8-8 data set. Finally, our 50-8-8 data set is multilingual, covering English (EN), German (DE), and Italian (IT). The three languages are challenging for word representations due to their large vocabulary, heavy reliance on word compounding (DE), and complex grammar and sentence structure (DE and IT).

In our paper, we contribute:

- **Reproduction studies** of outlier identification and word analogy (Camacho-Collados

* Equal contribution

Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems (Eval4NLP), pages 120–130,
November 20, 2020.

and Navigli, 2016; Köper et al., 2015; Berardi et al., 2015; Mikolov et al., 2013a) through which we find that most evaluations are reproducible, albeit some, namely outlier identification, only after taking variance into account.

- **50-8-8**, an improved outlier identification data set that addresses issues with the 8-8-8 data set used in the original outlier identification paper. 50-8-8 is multiple times larger than 8-8-8, multilingual (English, German, and Italian), excludes polysemous and rare words, and contains both semantic and syntactic tests.

- **Comparative study** and analysis of CBOW, skip-gram, word2vecf, and word2vecf without relation-suffixes, on multiple corpora and languages (English, German, and Italian), for multiple hyper-parameters, on outlier identification and analogy reasoning tasks (both semantic and syntactic). All results are based upon multiple instances of the models and quantify variation in results.

2 Related Work

Contextualized neural word embeddings (Devlin et al., 2019; Liu et al., 2019) show impressive performance in downstream NLP tasks, at the cost of training time; pre-training of the base version of BERT took four days using 16 TPU chips (Devlin et al., 2019). Efforts to reduce the training time still require significant computing power on dedicated hardware (You et al., 2020), with high environmental cost (Strubell et al., 2019). Some reduction of memory usage (Sanh et al., 2019) or of training time and memory usage (Lan et al., 2020) still does not eliminate the high resource consumption. As such, less computationally expensive models, such as word2vec (Mikolov et al., 2013a), word2vecf (Levy and Goldberg, 2014a), FastText (Bojanowski et al., 2017), and GloVe (Pennington et al., 2014), are attractive when showing good performance on NLP tasks.

Computationally cheaper models, like word2vec, have some of the same evaluation drawbacks as their more complicated and expensive counterparts: there is no generally agreed upon evaluation. Ghannay et al. (2016) compare word2vec and word2vecf on attributional similarity, extended by Li et al. (2017) for combinations of context representations and context types for CBOW, skip-gram, and GloVe. But, Faruqui et al. (2016) and Batchkarov et al. (2016) note that attributional similarity is subjective, lacks statistical significance, and has a low correlation with extrinsic evaluation, making it inconsistent and not necessarily indicative of model properties. However, Schnabel et al. (2015) argue that different extrinsic evaluation tasks prefer different embeddings, suggesting that extrinsic tasks might not be indicators of general embedding quality either.

The outlier identification task (Camacho-Collados and Navigli, 2016) avoids subjective similarity measurements. Instead, it employs relative word vector similarity to identify an outlier from a group of otherwise semantically related words. Blair et al. (2017) expanded the outlier identification data set algorithmically based on Wikidata. However, the automatic approach has several limitations, including ambiguous, infrequent, or duplicate words in the same category, and word variants in the same category, likely due to hierarchy inconsistencies in Wikidata (Brasileiro et al., 2016). In this paper, we return to manually curated data sets with controlled quality and difficulty.

In light of recent revelations into the instability of word2vec (Antoniak and Mimno, 2018), we reproduce several word vector evaluations. We find that the original 8-8-8 data set used in the outlier identification evaluation leads to high results variance. We address this issue by proposing an expanded evaluation data set we call 50-8-8. Both the original outlier identification (Camacho-Collados and Navigli, 2016) and word similarity publications (Ghannay et al., 2016; Li et al., 2017) do not fully explore the effects of hyper-parameters and randomness. We systematically evaluate models and hyper-parameters on ten training runs and measure average performance and variance.

Finally, most evaluations of word2vec embeddings focus on English, with notable exceptions (Köper et al., 2015; Berardi et al., 2015; Svoboda and Brychcín, 2018; Venekoski and Vankka, 2017; Rodrigues et al., 2016; Chen et al., 2015; Grave et al., 2018). However, these are translations of word similarity tasks and share the weaknesses of their English language counterparts. We reproduce the evaluation of core word analogy evaluations of Köper et al. (2015) and Berardi et al. (2015) and expand them by comparing word2vec to its dependency-based counterpart, word2vecf. We use the word analogy task from Mikolov et al. (2013a) to give a reference point for model performance

and ease comparison with other research, even though the pitfalls from the similarity tasks also apply to this task (Faruqui et al., 2016; Batchkarov et al., 2016). To supplement the evaluations on non-English languages, we manually translate our new 50-8-8 data set into German and Italian and thus provide a multilingual outlier identification data set and evaluation.

3 Tasks

In this section, we introduce the intrinsic tasks and data sets we use for evaluation. Furthermore, we summarize previous data sets' limitations and introduce a new data set for the outlier identification task.

3.1 Outlier identification

Evaluations of word similarity rely on a similarity score of words. Therefore it is difficult (if not impossible) to obtain a gold standard as people cannot agree on similarity scores between words (e.g., Which is more like a *cat*? a *tiger* or a *lion*?). On the other hand, outlier identification aims to identify an outlier in a set of similar words. The outlier is the word with the lowest average cosine similarity to the rest of the set. This formulation makes constructing a gold standard more straightforward as the attribution of specific similarity scores is avoided (Camacho-Collados and Navigli, 2016). Even though word embeddings cannot answer questions involving subtle similarity, they can represent outliers as sufficiently distinctive from a group of words that share some similarities (the inliers).

3.1.1 Measures for outlier identification

We use two performance measures to evaluate, **Accuracy** (Acc) and **Outlier Position Percentage** (OPP). Accuracy is the ratio of correctly identified outliers to the total number of test cases and provides a strict, narrow-focused measure of performance. OPP indicates how close the outliers are to being correctly classified. OPP is defined as:

$$OPP = \frac{\sum_{W \in D} \frac{OP(W)}{|W|-1}}{|D|} \cdot 100$$

W is a word set (8 inliers and one outlier), and D is a data set consisting of $|D|$ such sets of words. Outlier Position (OP) is the outlier position in the list of words ordered by the average cosine similarity to the other words in the set. The positions range from 0 to $|W| - 1$, where an OP equal to $|W| - 1$ indicates a correct classification of the outlier, and a lower OP indicates the computed position of the outlier in the sorted list. The lower the OP, the worse the system does at identifying the outlier. While accuracy takes a black-and-white approach to measuring performance, OPP accounts for differences in the words' rankings.

For our experiments, we modify the original evaluation script of Camacho-Collados and Navigli to address a bug. In the script, vectors are set to the zero vector for Out-Of-Vocabulary (OOV) words, resulting in an undeserved successful outlier identification. In our experiments, we instead mark such test cases as unsuccessful. Accordingly, OOV words decrease performance scores instead of increasing it. We describe the error and our fix in Appendix A and share our fixed script with our 50-8-8 data set[4].

3.1.2 Data sets for outlier identification

Camacho-Collados and Navigli (2016) provide a manually curated **8-8-8 data set** with their task; namely, 8 test groups of 8 semantically related inliers and 8 alternatives for non-related outliers, resulting in 64 test cases. The data set, however, has some limitations. First of all, its low number of test cases results in a significant change in accuracy for each misclassification. The low number also results in limited coverage of concepts in a vector space, which may not represent the semantic information encoded. Secondly, it contains ambiguous words. For example, *Smart* (used in the *German car manufacturers* test group) can denote both the car manufacturer and an unrelated adjective. Because the adjective might be more common in a corpus, it will have a higher influence on the resulting vector and might lead to its corresponding word being classified as an outlier. We claim that selecting the word "Smart" as an outlier when the adjective is prevalent is, in fact, the correct behavior. However, since this goes against the intention of the data set design (and the ground-truth labels), we consider such ambiguous words a drawback. Thirdly, multi-token words are handled by taking the average vector of all constituting tokens, which is problematic. The concept denoted by a multi-token word does not necessarily have connections to the meaning (i.e., vector) of the tokens that comprise it.[1] Finally, some words in the data set have a

[1] Mercedes Benz comprises two proper names. Mercedes

very low frequency in the corpora used for training in the original paper.[2] Low-frequency terms tend to have unstable word vectors, which can lead to high variance in evaluation using the 8-8-8 data set.

WikiSem500 (Blair et al., 2017) is an automatically generated extension of 8-8-8. By treating Wikidata as a graph such that semantic word similarities are distances in the graph, the authors of WikiSem500 automatically construct 500 test groups and 2 816 test cases. However, WikiSem500 has severe limitations. First of all, many inlier sets have a vague semantic connection that makes outliers difficult to identify (even for humans), which may be caused by Wikidata not always following structural rules from multi-level model theory (Brasileiro et al., 2016). WikiData's crowd-sourced nature causes many hierarchies spanning more than one classification level to follow known anti-patterns such as items that are simultaneously instances and subclasses other items; items that are subclasses of several items, with one of the superclasses an instance of the other, and lastly, items representing instances of several items, with one of those also an instance of the other (Brasileiro et al., 2016). Such inconsistencies in the graph are reflected in some of the test groups in WikiSem500. Take, for example, test group Q197, which consists of instances of airplanes. The inliers include various specific combat aircraft models (e.g., *B-29_Superfortress* and *F/A-18_Hornet*) and also the terms *glider* and *fighter_aircraft*, which should be subclasses rather than instances of airplanes and should therefore not be inliers. At the same time, *Mitsubishi F-1* (a Japanese combat aircraft) is an outlier, although it should be an instance of an airplane, and therefore an inlier.

Other problems include: ambiguous words, the same outliers appear several times in the same test group (thus overly impacting evaluation results), the same words with different spellings in the same test group, infrequent words, and inconsistency between using the same words or new ones in the same test group for different languages[3].

To overcome the above issues with 8-8-8 and WikiSem500, we propose **50-8-8**[4], a manually curated data set comprising two sections: **25-8-8-Sem** and **25-8-8-Syn**. We select unambiguous single-token[5] words with a minimum frequency of 350 in each training corpus (details in Section 4.2). We determine word ambiguity using dictionaries and native speakers. Our outliers have different degrees of connectedness to the inliers for different levels of test complexity, i.e., the further down the list of outliers, the weaker the connection to the inliers, and more evidently an outlier.

For example, in the test group *Greek Gods*, the first two outliers are *Cupid* (Roman god of love) and *Odysseus* (Greek legendary king), which could be misclassified by someone with little domain knowledge. The following are *Jesus*, *Sparta*, *Delphi*, and *Rome*, all of which have only a weak connection to the inliers. The last two outliers are *wrath* and *Atlanta*, with no connection to the inliers. **25-8-8-Sem** contains 25 test groups, each comprising eight inliers and eight alternatives for outliers, resulting in 200 unique test cases, a more than 3-fold increase in size over the original 8-8-8 data set.

Please note that in preliminary experiments, we found that random selection of outliers produces trivial test cases, with all models scoring above 97.05 in accuracy and 99.15 in OPP.

The second part of our *50-8-8* data set, the syntactic **25-8-8-Syn** data set consists of 25 syntactic test groups, as defined by part-of-speech tags (PoS). We choose words with a unique PoS tag in dictionaries to avoid syntactic ambiguity[6]. Furthermore, we ensure that the words in each test case share no semantic connection, such that evaluation can focus exclusively on distinction by syntactic role.

The two distinct subsets of **50-8-8** improve the outlier identification task by allowing for evaluations that target semantics and syntax, the two core aspects that word vectors encode.

In addition to English, we also look at **German** (another West Germanic language) and **Italian** (a Romance language), which both employ a more complex grammatical structure than English, and use declension to mark gender and plurality. German also relies heavily on compound words and grammatical cases. We manually translate our 50-

is a popular female name in latin-language countries, not related to cars like Mercedes Benz.

[2]E.g. Nestlé, Thaddaeus, and Alpina have a frequency of 17, 24, and 27, respectively, in UMBC.

[3]e.g. Q9143, Q341, Q16970, Q23691, and Q349, respectively.

[4]The 50-8-8 data set is available for download at https://github.com/JesperBrink/50-8-8

[5]Except in special cases as explained in Appendix B

[6]There are minor differences in the definition of syntactic ambiguity, as explained in Appendix B

8-8 data set using dictionaries and native speakers. We address translation and language-specific challenges as follows. First of all, words that are unambiguous in one language can be ambiguous in another. We address semantic ambiguity by replacing ambiguous words in any language with words that are unambiguous in all languages, and syntactic ambiguity by replacing the ambiguous word with one belonging to the same PoS tag. Syntactic ambiguity is language-specific, e.g., when translating adverbs to German, as the suffixes *-ly* and *-mente* often distinguish adjectives from adverbs in English and Italian, respectively, but German can use the same lexical form for both[6]. In Italian, many adjectives are also nouns, and many nouns are also conjugations of verbs, which are not as prevalent in German and English. Secondly, when a word translates to two synonymous words, we use the most common, as determined by native speakers.

Furthermore, for nouns in German, we use the nominative case of the nouns to avoid the effects of different grammatical cases. For adjectives in Italian, we use the masculine gender where applicable to avoid the effects of gender. Removing syntactic variation allows the semantic tests to stay focused on semantics. Thus, all the versions of 25-8-8-Sem are identical, all versions of 25-8-8-Syn have an identical distribution of PoS tags within a given test group, and we use consistent and frequent variants of words.

3.2 Word analogy task

Our study's second task is the word analogy task, which measures how well a model captures the relational similarity between pairs of words. A high degree of relational similarity between the pairs means that the words are analogous (Mikolov et al., 2013c; Turney, 2006). It includes questions like *Berlin is to Germany as what is to France?* where the model should return *Paris*. Word analogy also has separation into semantic and syntactic tests. As we note in Section 2, there is heavy criticism of this task(Faruqui et al., 2016). We include it for easy comparison with existing work and to contextualize the outlier identification results.

3.2.1 Data set for word analogy task

We use the analogy data set of Mikolov et al. (2013a) consisting of 19 544 test cases in 14 different categories capturing different relations, nine syntactic and five semantic, resulting in 10 675 syntactic and 8 869 semantic test cases. For German, we use a version of the analogy data set, which has a total of 18 552 test cases (the adjective-adverb category is missing as it does not exist in German) (Köper et al., 2015). We use an Italian translation of the analogy data set (Berardi et al., 2015), with 19 791 test cases, with small changes to the data set to keep all words as single token words.

Please note that the word analogy data set is not balanced. Size varies by category, causing some relations to be over-represented, e.g., two of the semantic categories evaluate knowledge about countries and corresponding capitals and represent more than half of the total semantic tests (Gladkova et al., 2016).

Corpus	Corpus length	Vocab. size
UMBC	3 457 177 447	1 465 802
Wiki EN	2 571 028 591	2 306 628
Wiki DE	896 693 693	2 154 939
Wiki IT	541 134 131	806 992

Table 1: Summary of corpora

4 Models and corpora

This section introduces the word embedding models and the training corpora we use for the evaluation.

4.1 Models

Word2vec consists of two types of models: CBOW (continuous-bag-of-words) and skip-gram (Mikolov et al., 2013a,b). Both models use a linear context, consisting of the n words before and n words after the current word.

Word2vecf (Levy and Goldberg, 2014a) replaces the linear context with one based on words directly connected via the dependency graph of the sentence. Thus, word2vecf eliminates the window size hyper-parameter of word2vec, increases the pool of available context tokens up to the sentence boundaries, and focuses context words selection by eliminating irrelevant words. The example *Australian scientist discovers star with a telescope* from the original paper can help understand the difference in context. For the word *discovers* and a window size of 2, word2vec would consider the words *Australian*, *scientist*, *star*, and *with* to be part of the context. There is nothing inherently Australian about discovering; hence, this word and *with* provide noise to the context of *discovers*. Word2vecf, instead, includes *scientist_nsubj*,

star_obj, and *telescope_prepwith* into the context. Thus, word2vecf both removes noisy words (*australian*, *with*) and includes relevant terms (*telescope*) into the context.

On the downside, word2vecf requires the corpus to be dependency parsed using a dependency parser, introducing some noise (Chen and Manning, 2014). Word2vecf suffixes the dependency relation to each word in the context, which massively increases vocabulary size up to $|V| \cdot |D|$ where $|V|$ denotes the vocabulary size and $|D|$ denotes the number of relation types supported by the dependency parser. The massive vocabulary increase leads to lower frequency counts and can result in instability in vectors' values. Furthermore, the word vectors are trained on the auxiliary words with the relation as suffix instead of training word vectors directly on each other, and as such, words with dependency relations suffixed act as barriers to information flow between context words and target words.

Word2vecf+ addresses the limitations of word2vecf, more specifically the inclusion of dependency relations as word suffixes; thus, the vocabulary size does not increase. Word2vec+ maintains the vocabulary size fixed by removing the suffix from the word before training, thereby training words directly on each other and discarding the auxiliary words (Li et al., 2017). For example, the word *scientist_nsubj* from above becomes *scientist*. While the original paper calls this method *generalized skip-gram with unbound dependency-based context*, for readability, we refer to it as word2vecf+.

4.2 Training Corpora

We use multiple corpora to derive word vectors for our evaluations (see summary in Table 1). For English, we use the UMBC web-based corpus (Han et al., 2013) and the September 2019 dump of English Wikipedia. The choice of corpora aims to reproduce the experiments in the original outlier identification work (Camacho-Collados and Navigli, 2016). The newer version of Wikipedia is a super-set of the one used in the original experiments. As in the original paper, the use of two English corpora should eliminate questions of corpus-specific results.

For German, we derive vectors from the January 2020 version of Wikipedia; for Italian from the April 2020 version of Wikipedia. The three versions of Wikipedia have widely different sizes. The largest (Wiki EN) is almost five times bigger than the smallest (Wiki IT). However, even the smallest has over 500 million tokens for a vocabulary of less than one million word types (average word type frequency of 670). The smallest corpus (Wiki IT) has a larger average word type frequency (670) than the second smallest, Wiki DE (average word type frequency 416). Such large corpora, combined with repetitions of training and evaluation cycles, provide a good overview of model performance and avoid the of word2vec (Antoniak and Mimno, 2018).

We use WikiExtractor (Attardi, 2018) to extract plain text from the Wikipedia corpora, and tokenize all corpora using Stanford CoreNLP v3.9.2 (Manning et al., 2014). We remove words that appear less than five times using the original word2vec code (Mikolov et al., 2013a) or word2vecf (Levy and Goldberg, 2014a), as appropriate. We dependency parse using the Stanford neural-network dependency parser for models that require dependency relations (word2vecf, word2vecf+) (Chen and Manning, 2014). For dependency parsing Italian, we use the model trained by Palmero Aprosio and Moretti (2016).

5 Experiments

This section presents experimental results, focusing on the reproduction, new data set, window size, different corpora, and languages. In our tables and figures, we denote the different approaches as follows: **CBOW**, **SG** (skip-gram), **W2VF** (word2vecf), **W2VF+** (word2vecf+); each followed by the size of the window used. We include a detailed description of the experimental setup in Appendix C.

5.1 Reproduction results

In Table 2, we compare our reproduction results with those of Camacho-Collados and Navigli (2016). We observe a high variance in accuracy, which illustrates the small 8-8-8 data set's weakness and further underlines the importance of evaluating multiple training runs. We conclude that the original outlier identification results can be reproduced, but with the caveat that accuracy can suffer from large variance. In Section 5.2, we propose 50-8-8, a data set that alleviates this issue.

Table 3 shows the results of reproducing the

Model	Work	UMBC OPP	UMBC Acc	Wiki OPP	Wiki Acc
CBOW 5	Original	93.80	73.40	95.30	73.40
	Our	93.69 ± 0.11	71.88 ± 4.39	94.51 ± 0.09	67.66 ± 1.00
SG 10	Original	92.60	64.10	93.80	70.30
	Our	92.75 ± 0.20	62.81 ± 5.27	94.16 ± 0.05	69.53 ± 1.10
CBOW 2	Our	93.38 ± 0.06	67.97 ± 2.08	94.94 ± 0.04	68.13 ± 1.07
CBOW 10	Our	94.10 ± 0.10	**72.34** ± 2.47	94.41 ± 0.02	68.13 ± 1.07
SG 2	Our	**94.61** ± 0.04	69.53 ± 1.59	**95.41** ± 0.07	71.72 ± 1.68
SG 5	Our	94.16 ± 0.02	69.84 ± 0.51	94.59 ± 0.08	69.69 ± 2.54
W2VF	Our	89.43 ± 0.06	62.50 ± 0.00	92.83 ± 0.20	68.75 ± 0.00
W2VF+	Our	92.46 ± 0.05	66.41 ± 0.61	94.47 ± 0.04	**75.78** ± 1.10

Table 2: Outlier identification reproduction of Camacho-Collados and Navigli (2016) (10 runs, 8-8-8 data set); word2vec with different window sizes, word2vecf and word2vecf+ added for easier comparison with other results.

word analogy task[7,8]. For English, we see the same pattern as (Mikolov et al., 2013a), where skip-gram outperforms CBOW, even though our corpora and hyper-parameters differ. Comparing the German Wikipedia results to those of (Köper et al., 2015), we see a similar pattern in the semantic part, where skip-gram outperforms CBOW. However, in the syntactic part, our results differ. Köper et al. observe that CBOW outperforms skip-gram, whereas we observe the opposite, which could be due to the difference in corpora and hyper-parameters such as vector dimensionality.

Due to the different focus of this paper and that of Berardi et al. (2015), we can only compare skip-gram results with window size 10. We observe a similar semantic performance, but a significant difference in syntactic performance where Berardi et al. observe a score of 32.62 compared to our result of 44.63, which could be the result of the difference in the number of negative samples (we use 15, they use 10) and the different Wikipedia version. However, as they do not cover the CBOW model, it is difficult to get an overview of model performance.

5.2 The effect of the new 50-8-8 data set

The results of outlier identification using our proposed 50-8-8 are in Table 4. As expected, given the more comprehensive tests, on both UMBC and English Wikipedia, we see significantly lower accuracy variance for 25-8-8-Sem than 8-8-8. The only exception is word2vecf, where the accuracy variance grows slightly from 0 on 8-8-8 up to 0.15 on 25-8-8-Sem. Although word2vecf accuracy variance on 8-8-8 is 0, the ten instances do differ in their answers, as can be observed in the OPP variance in Table 2. Except for a few individual cases, the variance on 25-8-8-Syn is also low. The performance of the best models on 25-8-8-Syn usually matches that on 25-8-8-Sem, suggesting that the two subsets of 50-8-8 are balanced in terms of difficulty. The best performing models on 25-8-8-Syn is CBOW 2 (except for Italian).

5.3 Effect of window size

Table 4 shows that window size has a limited impact on OPP for semantic tests (25-8-8-Sem), but affects the results on syntactic tests (25-8-8-Syn), where skip-gram performs best with low window size across all corpora. For the word analogy task (Table 3), the opposite is true for the semantic evaluation, where larger window sizes have improved performance. These results align with Bansal et al. (2014), who observe that larger window sizes result in more semantic information, while smaller lead to more syntactic.

The same pattern can be observed on syntactic German Wikipedia and syntactic UMBC when taking variance into account. Bansal et al. observe that CBOW and skip-gram with lower window size perform better on syntactic tests, and larger window size performs better on semantic tests. However, our results show that window size performance varies with the task. These two tasks' preferred window sizes indicate that lower window sizes better capture clusters with semantically and syntactically similar words. Larger window sizes are better suited for capturing word relations. These observations also indicate that hyper-parameters can have a big influence on the performance of the models.

5.4 Effect of context type

Table 4 casts a shadow on the superiority of the word2vecf context construction strategy. Word2vecf matches or trails the best word2vec

[7]Note that 5% of the questions were skipped by the German models and 10% of the questions were skipped by the Italian models due to OOV words. This was also observed by Berardi et al. (2015).

[8]We use the 3CosAdd method for solving the task, just like (Mikolov et al., 2013a). The alternative 3CosMul improves the analogy results and is discussed in Appendix D.

Model	UMBC Sem	UMBC Syn	EN Wiki Sem	EN Wiki Syn	DE Wiki Sem	DE Wiki Syn	IT Wiki Sem	IT Wiki Syn
CBOW 2	10.37 ± 0.03	51.92 ± 0.05	25.34 ± 0.09	43.92 ± 0.03	16.38 ± 0.07	15.28 ± 0.06	4.38 ± 0.02	21.42 ± 0.08
CBOW 5	15.96 ± 0.06	53.01 ± 0.04	35.17 ± 0.17	48.18 ± 0.06	22.36 ± 0.11	17.70 ± 0.07	5.11 ± 0.02	26.06 ± 0.02
CBOW 10	23.36 ± 0.05	54.47 ± 0.06	51.75 ± 0.02	50.95 ± 0.02	27.01 ± 0.11	18.40 ± 0.04	6.57 ± 0.03	28.06 ± 0.07
SG 2	56.29 ± 0.58	68.72 ± 0.07	72.84 ± 0.09	63.74 ± 0.09	53.93 ± 0.10	28.45 ± 0.05	28.06 ± 2.62	42.77 ± 0.11
SG 5	64.59 ± 0.13	**69.51** ± 0.07	77.70 ± 0.14	**64.36** ± 0.04	66.26 ± 0.29	31.63 ± 0.08	44.01 ± 0.07	**44.98** ± 0.07
SG 10	**67.59** ± 0.56	69.19 ± 0.80	**78.42** ± 0.04	62.36 ± 0.09	**68.15** ± 0.08	**32.15** ± 0.03	**50.77** ± 0.17	44.63 ± 0.12
W2VF	9.39 ± 0.08	54.75 ± 0.16	15.22 ± 0.22	46.05 ± 0.03	6.40 ± 0.02	12.30 ± 0.03	2.3 ± 0.01	21.38 ± 0.01
W2VF+	30.63 ± 0.23	65.82 ± 0.03	51.90 ± 0.35	62.76 ± 0.03	19.41 ± 0.17	24.49 ± 0.06	7.10 ± 0.07	33.41 ± 0.18

Table 3: Word Analogy on all training corpora; model name followed by window size.

Corpus	Model	25-8-8-Sem OPP	25-8-8-Sem Acc	25-8-8-Syn OPP	25-8-8-Syn Acc
UMBC	**CBOW 2**	95.67 ± 0.01	85.85 ± 0.30	**94.38** ± 0.02	73.75 ± 0.06
	CBOW 5	95.67 ± 0.40	85.50 ± 0.35	94.31 ± 0.02	75.85 ± 0.15
	CBOW 10	95.57 ± 0.10	84.75 ± 0.31	93.98 ± 0.04	75.15 ± 0.35
	SG 2	96.83 ± 0.40	87.00 ± 0.40	92.48 ± 0.13	73.35 ± 0.95
	SG 5	96.79 ± 0.01	86.15 ± 0.05	86.90 ± 0.15	62.40 ± 1.29
	SG 10	96.68 ± 0.03	86.40 ± 0.49	82.86 ± 0.24	53.55 ± 2.17
	W2VF	96.09 ± 0.03	84.65 ± 0.15	94.15 ± 0.38	**80.35** ± 1.95
	W2VF+	**97.41** ± 0.01	**89.45** ± 0.32	91.54 ± 0.70	71.55 ± 4.47
Wiki EN	**CBOW 2**	95.83 ± 0.01	83.60 ± 0.44	**95.53** ± 0.02	80.00 ± 0.50
	CBOW 5	96.14 ± 0.02	85.00 ± 0.20	95.26 ± 0.01	**80.30** ± 0.66
	CBOW 10	95.74 ± 0.01	83.10 ± 0.09	94.55 ± 0.02	77.05 ± 0.27
	SG 2	**97.68** ± 0.01	**88.75** ± 0.41	90.09 ± 0.25	67.00 ± 1.65
	SG 5	97.44 ± 0.01	88.50 ± 0.00	86.33 ± 0.69	59.60 ± 1.54
	SG 10	97.05 ± 0.01	87.45 ± 0.07	82.74 ± 1.01	54.00 ± 1.75
	W2VF	94.66 ± 0.03	80.45 ± 0.12	90.63 ± 0.11	70.00 ± 0.75
	W2VF+	97.07 ± 0.00	88.00 ± 0.10	84.29 ± 0.03	54.75 ± 0.41
Wiki DE	**CBOW 2**	92.41 ± 0.05	74.60 ± 1.34	**93.76** ± 0.08	**72.95** ± 1.97
	CBOW 5	92.24 ± 0.04	73.40 ± 0.44	92.46 ± 0.03	68.95 ± 1.02
	CBOW 10	92.48 ± 0.03	72.65 ± 0.70	91.65 ± 0.05	65.70 ± 0.51
	SG 2	**93.93** ± 0.04	**79.25** ± 0.16	89.16 ± 0.03	64.05 ± 1.22
	SG 5	**93.93** ± 0.01	78.00 ± 0.70	86.09 ± 0.12	55.40 ± 1.14
	SG 10	93.83 ± 0.03	76.90 ± 0.29	83.41 ± 0.43	51.55 ± 2.02
	W2VF	91.40 ± 0.00	69.60 ± 0.24	90.28 ± 0.02	72.80 ± 0.26
	W2VF+	93.29 ± 0.01	74.55 ± 0.57	78.58 ± 0.14	48.85 ± 0.50
Wiki IT	**CBOW 2**	94.29 ± 0.06	75.05 ± 0.62	93.41 ± 0.04	75.10 ± 0.84
	CBOW 5	93.88 ± 0.05	73.15 ± 0.40	93.98 ± 0.05	76.55 ± 1.67
	CBOW 10	93.21 ± 0.04	71.55 ± 0.62	93.77 ± 0.04	74.55 ± 2.17
	SG 2	**95.44** ± 0.06	**79.15** ± 0.45	81.20 ± 0.26	60.75 ± 0.51
	SG 5	95.13 ± 0.02	77.40 ± 0.39	78.69 ± 0.17	56.20 ± 0.91
	SG 10	94.93 ± 0.02	77.40 ± 0.19	75.51 ± 0.09	50.10 ± 0.29
	W2VF	92.48 ± 0.01	70.20 ± 0.06	**95.37** ± 0.02	**83.80** ± 0.56
	W2VF+	94.39 ± 0.03	75.40 ± 0.84	78.59 ± 2.15	55.10 ± 2.14

Table 4: Outlier identification on 50-8-8 (25-8-8-Sem, 25-8-8-Syn); model name followed by window size.

model on semantic tests on all corpora. However, word2vecf seems better suited to syntactic tests, where it matches or outperforms the best word2vec model on all four corpora.

We observe the same results in the word analogy task (Table 3). Despite the expected improvements in the contexts of word2vecf and word2vecf+, they consistently underperform the word2vec models, sometimes underperforming even the weakest of the word2vec models. This observation is consistent across all data sets on all languages.

5.5 Effect of relation-suffix

The results in Table 4 show that word2vecf+ outperforms word2vecf on semantic outlier identification across all corpora. On the syntactic subset, 25-8-8-Syn, word2vecf consistently outperforms word2vecf+ on all corpora. The consistent difference in performance between word2vecf and word2vecf+ on both the semantic and syntactic tests suggests that word2vecf might be better suited for encoding syntactic information and word2vecf+ might be better suited for encoding semantic information.

We observe a large drop in syntactic OPP and accuracy for both word2vecf and word2vecf+ from UMBC to Wiki EN. The drop may be due to the quality of dependency relations from the Stanford CoreNLP dependency parser, which learned from the Penn Treebank, a corpus of scientific abstracts, news stories, and bulletins (Chen and Manning, 2014; Marcus et al., 1993). Thus, Penn Treebank resembles UMBC more than English Wikipedia, which could explain the performance drop.

On the word analogy task (Table 3), word2vecf+ performs better than word2vecf. On the syntactic tests, word2vecf is comparable to CBOW, but removing the relation suffix (word2vecf+) results in scores closer to skip-gram, which is the best performing model; on the semantic tests, removing

the relation suffix results in a 3-fold increase in word2vecf+ performance over word2vecf.

Based on these observations, we conclude that word2vecf+ is better able to capture semantic information as it avoids word2vecf's dramatic, artificial, increase in vocabulary. It allows word vectors to directly influence each other during training resulting in better semantically positioned related words in the embedding space and better capturing both syntactic and semantic similarities in word pairs. In contrast, the relational suffixes improve the clustering of syntactically related words.

5.6 Results across languages

Table 4 shows that the models trained on German and Italian are generally less capable than those trained on the English corpora. The difference between German and English is noticeable in syntactic analogy (Table 3). The German performance is almost half that of English across all models while Italian is better, but is still significantly lower than English. Furthermore, in the semantic part of word analogy, the performance of models trained on UMBC is closer to models trained on Wiki DE than models trained on Wiki EN. In general, Table 4 shows a drop in performance for languages other than English, in line with our expectation that German and Italian are more difficult to model.

6 Conclusions

We contribute several reproduction studies of the outlier identification task and the classic word analogy task, both intrinsic evaluations of non-contextual word representations. We provide an in-depth analysis of *word2vec*, *word2vecf*, and *word2vecf+* on the two tasks analyzing the effects of window size, context type, and context representation on English, German, and Italian. We find that the context construction strategy of word2vecf and word2vecf+ is not always effective. Sometimes the two models underperform even the weakest of the word2vec models.

Our reproduction of outlier identification shows high variance, which we attribute to the original data set's limitations. To address these limitations, we propose 50-8-8, a new data set that is multiple times larger, manually curated, multilingual, and contains syntactic and semantic tests. Besides eliminating the variance issues, 50-8-8 quantifies the drop in performance in representations of languages with more complicated grammar and morphology than English.

Acknowledgments

We would like to thank Davide Mottin for helping with the translation of 50-8-8 to Italian.

References

Maria Antoniak and David Mimno. 2018. Evaluating the stability of embedding-based word similarities. *Transactions of the Association for Computational Linguistics*, 6:107–119.

Giuseppe Attardi. 2018. Wikiextractor. https://git.io/fARaC.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 809–815, Baltimore, Maryland. Association for Computational Linguistics.

Miroslav Batchkarov, Thomas Kober, Jeremy Reffin, Julie Weeds, and David Weir. 2016. A critique of word similarity as a method for evaluating distributional semantic models. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 7–12, Berlin, Germany. Association for Computational Linguistics.

Giacomo Berardi, Andrea Esuli, and Diego Marcheggiani. 2015. Word embeddings go to italy: A comparison of models and training datasets. In *Proceedings of the 6th Italian Information Retrieval Workshop, Cagliari, Italy, May 25-26, 2015*, volume 1404 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Philip Blair, Yuval Merhav, and Joel Barry. 2017. Automated generation of multilingual clusters for the evaluation of distributed representations. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*.

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.

António Branco, Nicoletta Calzolari, Piek Vossen, Gertjan Van Noord, Dieter van Uytvanck, João Silva, Luís Gomes, André Moreira, and Willem Elbers. 2020. A Shared Task of a New, Collaborative Type to Foster Reproducibility: A First Exercise in the Area of Language Science and Technology with REPROLANG2020. pages 5541–5547, Marseille, France. European Language Resources Association.

António Branco, Kevin Bretonnel Cohen, Piek Vossen, Nancy Ide, and Nicoletta Calzolari. 2017. Replicability and reproducibility of research results for human language technology: introducing an LRE special section. *Language Resources and Evaluation*, 51(1):1–5.

Freddy Brasileiro, João Paulo A. Almeida, Victorio A. Carvalho, and Giancarlo Guizzardi. 2016. Applying a multi-level modeling theory to assess taxonomic hierarchies in wikidata. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, page 975–980, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

José Camacho-Collados and Roberto Navigli. 2016. Find the word that does not belong: A framework for an intrinsic evaluation of word vector representations. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 43–50, Berlin, Germany. Association for Computational Linguistics.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar. Association for Computational Linguistics.

Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huan-Bo Luan. 2015. Joint learning of character and word embeddings. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1236–1242. AAAI Press.

Drummond Chris. 2009. Replicability is not Reproducibility: Nor is it Good Science. In *The 4th workshop on Evaluation Methods for Machine Learning held at ICML 2009*, Montreal, Canada.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 30–35, Berlin, Germany. Association for Computational Linguistics.

Sahar Ghannay, Benoit Favre, Yannick Estève, and Nathalie Camelin. 2016. Word embedding evaluation and combination. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 300–305, Portorož, Slovenia. European Language Resources Association (ELRA).

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pages 8–15, San Diego, California. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning Word Vectors for 157 Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Lushan Han, Abhay L. Kashyap, Tim Finin, James Mayfield, and Jonathan Weese. 2013. UMBC_EBIQUITY-CORE: Semantic textual similarity systems. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 44–52, Atlanta, Georgia, USA. Association for Computational Linguistics.

Maximilian Köper, Christian Scheible, and Sabine Schulte im Walde. 2015. Multilingual reliability and "semantic" structure of continuous word spaces. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 40–45, London, UK. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

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

Omer Levy and Yoav Goldberg. 2014b. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 171–180, Ann Arbor, Michigan. Association for Computational Linguistics.

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.

Bofang Li, Tao Liu, Zhe Zhao, Buzhou Tang, Aleksandr Drozd, Anna Rogers, and Xiaoyong Du. 2017. Investigating different syntactic context types and context representations for learning word embeddings. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2421–2431, Copenhagen, Denmark. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: The penn treebank. *Comput. Linguist.*, 19(2):313–330.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff 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*, pages 3111–3119. Curran Associates, Inc.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

A. Palmero Aprosio and G. Moretti. 2016. Italy goes to Stanford: a collection of CoreNLP modules for Italian. *ArXiv e-prints*.

Roger D Peng. 2011. Reproducible Research in Computational Science. *Science*, 334(6060):1226–1227.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

João Rodrigues, António Branco, Steven Neale, and João Silva. 2016. Lx-dsemvectors: Distributional semantics models for portuguese. In *Computational Processing of the Portuguese Language*, pages 259–270, Cham. Springer International Publishing.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *5th Workshop on Energy Efficient Machine Learning and Cognitive Computing (NeurIPS)*.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307, Lisbon, Portugal. Association for Computational Linguistics.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Lukáš Svoboda and Tomáš Brychcín. 2018. New word analogy corpus for exploring embeddings of czech words. In *Computational Linguistics and Intelligent Text Processing*, pages 103–114, Cham. Springer International Publishing.

Peter D. Turney. 2006. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416.

Viljami Venekoski and Jouko Vankka. 2017. Finnish resources for evaluating language model semantics. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 231–236, Gothenburg, Sweden. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2020. Large batch optimization for deep learning: Training bert in 76 minutes. In *International Conference on Learning Representations*.