Evaluating Semantic Vectors for Norwegian

Cathrine Stadsnes, Lilja Øvrelid, Erik Velldal
Department of Informatics, University of Oslo
cathrine.stadsnes@ceres.no,
{liljao,erikve}@ifi.uio.no

Abstract

In this article, we present two benchmark data sets for evaluating models of semantic word similarity for Norwegian. While such resources are available for English, they did not exist for Norwegian prior to this work. Furthermore, we produce large-coverage semantic vectors trained on the Norwegian Newspaper Corpus using several popular word embedding frameworks. Finally, we demonstrate the usefulness of the created resources for evaluating performance of different word embedding models on the tasks of analogical reasoning and synonym detection. The benchmark data sets and word embeddings are all made freely available.

1 Introduction

In recent years, vector space models that implement a distributional approach to modeling the meaning of words have become subject of increasing research interest in the natural language processing (NLP) community. The basic idea of distributional semantics is that words that tend to occur in similar contexts can be assumed to have similar meanings. Hence, the meaning of a word can be inferred from its observed contexts of use in corpora, i.e., large text collections. In particular so-called *word embeddings*, i.e., low dimensional dense vector encodings of meaning, have proved popular recently, not the least because they have proved useful for specifying the input layer to neural network models in NLP tasks, replacing traditional feature engineering and making it possible to take advantage of large amounts of unlabeled data.

As task-based or *extrinsic* evaluation can be expensive, it may be desirable to quantify performance properties of vector models prior to downstream use. There exists a range of benchmark data sets that facilitate such *intrinsic* evaluation of model performance for English (Finkelstein et al., 2002; Hill et al., 2015). In contrast, Norwegian remains an under-resourced language in the sense that many core NLP resources are still missing. This includes resources for evaluating distributional semantic models.

This article presents two benchmark data sets for evaluating models of semantic word similarity for Norwegian. The Norwegian Analogy Test Set was created by semi-automatically translating and adapting the existing Google analogies test set (Mikolov et al., 2013a) from English to Norwegian, for the task of analogical reasoning. The

This paper was presented at the NIK-2018 conference; see http://www.nik.no/.

Norwegian Synonymy Test Set, defined for evaluating the task of synonym detection, was created by extracting words and associated synonyms from a digital version of an existing Norwegian synonym dictionary, *Norske synonymer blå ordbok*, published by Kunnskapsforlaget. In the following, we first describe the tasks of analogical reasoning and synonym detection in more detail, along with relevant previous work. We then go on to describe the creation of the Norwegian benchmark data sets, before finally using them to evaluate a range of different distributional semantic vector models. In particular, we attempt to isolate the effects of various modeling choices like text pre-processing and word embedding framework. The experimental results demonstrate the usefulness of the created resources for ranking the relative performance of different word embedding models. As a by-product of the evaluation experiments, we make available large-coverage semantic vectors for Norwegian.[1]

2 Task definitions and previous work

This section presents the tasks of analogical reasoning and synonym detection in more detail, along with previous work on their use for evaluating distributional word vectors.

Analogical reasoning The task of analogical reasoning has been widely used for evaluating distributional semantic models, following the seminal work of Mikolov et al. (2013a,b). Mikolov et al. (2013b) showed that the learned word representations in fact capture meaningful syntactic and semantic regularities in the form of constant vector offsets between pairs of words sharing a particular relationship.

Benchmark data sets typically consist of lists of two word pairs that share a relation, of the form $\langle a{:}b, c{:}d \rangle$, such as $\langle$*boy*:*girl*, *brother*:*sister*$\rangle$ or $\langle$*easy*:*easier*, *big*:*bigger*$\rangle$. These tuples are to be read as; word *a* is to *b* as *c* is to *d*. The inference task presented to a distributional vector model is typically to correctly infer the word *d* as an unknown, given the other three, and performance is measured in terms of accuracy of correctly answered questions. For example, if we denote the vector for a word i as x_i, Mikolov et al. (2013b) found that one could identify d by first computing $y = x_b - x_a + x_c$ and then retrieve the word whose embedding vector has the greatest cosine similarity to y.

The most widely used analogy data set for English is the Google analogies test set proposed by Mikolov et al. (2013a), comprising a total of 19,544 analogy questions divided into semantic and syntactic subsets. Another analogy test set for English is the SemEval data set (Jurgens et al., 2012), containing semantic relations, each exemplified by a few word pairs, and for each relation the model must rank a set of target word pairs according to the degree to which the relation applies.

We describe the Google analogies test set in more detail in Section 3, together with our adaption of the data set to Norwegian.

Synonym detection Another common task used for evaluating distributional vector models is to automatically identify words that are semantically close. This can either be semantically *similar* words, like *car* and *vehicle*, or semantically *related* words, such as *car* and *gasoline*. A commonly used test set for this task is WordSim-353 (Finkelstein et al., 2002) which contains 353 English word pairs with human-assigned similarity scores. Model performance is measured in terms of correlation between the cosine

[1]This article provides a condensed version of Stadsnes (2018), which contains more details on the resources and evaluation experiments described here.

similarity scores for the word pairs computed by the model and the corresponding average human-assigned scores. Agirre et al. (2009) further split this data set into similarity and relatedness subsets. The more recent SimLex-999 data set (Hill et al., 2015) focuses on semantic similarity exclusively. This data set was created by mining the opinions of 500 annotators via the crowdsourcing platform Amazon Mechanical Turk.

Alternatively, model performance can be evaluated on the stricter task of identifying *synonyms*. For English, this has been done using standardized synonym tests, like the TOEFL (Test of English as a Foreign Language) data set of Landauer and Dumais (1997) which contains 80 multiple-choice questions consisting of one target word along with four synonym candidates. The task of the model is to correctly choose the candidate that is most similar to the target word. To solve this task, the models compute cosine similarity scores between each candidate word vector and the target word vector and select the one that gives the highest score. Model performance is then evaluated in terms of accuracy of correct answers. Similarly, Leeuwenberg et al. (2016) evaluate the task of synonym detection using synonyms from WordNet[2] for English and GermaNet.[3] The wordnet resources contain a structured collection of so-called *synsets*, where each synset is a set of words that are considered to be synonyms. In their evaluation Leeuwenberg et al. (2016) defined *precision* to be the proportion of correctly predicted synonym word pairs from all predictions and *recall* to be the proportion of correctly predicted synonym word pairs from all synonym word pairs that are present in the wordnet. Because synonymy is symmetric, they considered the word pairs (w_1, w_2) and (w_2, w_1) equivalent during evaluation.

In our work, we will define a test set for synonym detection for Norwegian, based on adapting an existing Norwegian synonym dictionary, detailed in Section 4.

3 The Norwegian analogy test set

In this section we describe our adaption of the Google analogies test set proposed by Mikolov et al. (2013a) from English to Norwegian. The original Google data comprises 8,869 semantic questions covering five types of semantic relationships and 10,675 syntactic questions covering nine types of syntactic relationships. The semantic analogies are typically about places, like ⟨*Athens*:*Greece*, *Baghdad*:*Iraq*⟩, and the syntactic analogies are typically about verb tenses or forms of adjectives, such as ⟨*dancing*:*danced*, *decreasing*:*decreased*⟩. See the first column of Table 1 for an overview of the relations.

Our adaptation of the Google analogies test set to Norwegian proceeded via an initial round of automatic translation using Google Translate, followed by manual error correction and post-processing. Since the analogy pairs do not provide any linguistic context, translation performance is quite poor and the translations had to be manually corrected. We may characterize these corrections according to the cause of error, e.g., linguistic differences between English and Norwegian and extralinguistic differences, such as differences in culture and geography.

Due to linguistic differences between English and Norwegian a number of analogy relations were modified. Morphological differences pertain to the types of inflections that a word may take on. In English, verbs are inflected for person and number (*says* vs. *say*) whereas this is not the case in Norwegian. Hence, these analogy questions would result in pairs of identical words, for ex-

[2] https://wordnet.princeton.edu/

[3] http://www.sfs.uni-tuebingen.de/GermaNet/

	Relation type	Questions	Word pair 1		Word pair 2	
Semantic	Common capital city	506	Athen	Hellas	Bagdad	Irak
	All capital cities	4,524	Abuja	Nigeria	Accra	Ghana
	Currency	866	Algerie	dinar	Angola	kwanza
	City-in-county	2,542	Hønefoss	Buskerud	Stord	Hordaland
	Man–woman	506	gutt	jente	bror	søster
Syntactic	Adjective-to-adverb	992	munter	muntert	hel	helt
	Opposite	600	akseptabelt	uakseptabelt	vitende	uvitende
	Comparative	1,190	dårlig	dårligere	stor	større
	Superlative	930	dårlig	dårligst	stor	størst
	Nationality adjective	1,599	Albania	albansk	Argentina	argentinsk
	Past tense	1,560	danser	danset	avtar	avtok
	Plural nouns	1,122	banan	bananer	fugl	fugler
	Present tense	870	avta	avtar	beskrive	beskriver

Table 1: Number of questions and examples of word pairs within each relation type in the Norwegian Analogy Test Set.

ample the singular–plural verb pairs ⟨*decrease*:*decreases*, *describe*:*describes*⟩ would translate to ⟨*avtar*:*avtar*, *beskriver*:*beskriver*⟩. We therefore replaced this relation type with infinitival–present tense pairs using the same verbs. For example, the analogy question ⟨*decrease*:*decreases*, *describe*:*describes*⟩ was replaced by ⟨*avta*:*avtar*, *beskrive*:*beskriver*⟩. The *Present participle* relation type (e.g., *crying*) was also removed from the adapted Norwegian data set, since the use of present participle in Norwegian is relatively uncommon. Another morphological difference between English and Norwegian is found in the morphology of adverbs. In Norwegian, there is no dedicated adverb ending (like the English *-ly*) and most adjectives are identical in form to the corresponding adverb, a phenomenon referred to as *syncretism*. Examples of such adjective–adverb pairs are *amazing amazingly* which translate to *utrolig utrolig* and *most mostly* translating to *mest mest*. Therefore, all such word pair instances were removed from the Norwegian data set. The same holds for words that have the same form in singular and plural in Norwegian, e.g., *child children* equals *barn barn*. These singular–plural noun pairs were also removed.

In the post-processing of the translated analogy questions, we also observed lexical semantic differences between English and Norwegian. For instance, in Norwegian there is no distinction between female and male grandchildren as there is in English, e.g., *granddaughter* and *grandson* in the *Man–woman* relation type are both translated to *barnebarn*, meaning simply *grandchild*. The gender relation cannot be inferred from the word pair *barnebarn barnebarn* and we thus removed these identical word pairs from the Norwegian data set. We also found examples of English words that could not be translated to a single Norwegian word. For instance, in English, words may be negated by adding negative prefixes, such as *un-*, *in-* and *dis-*, to nouns, adjectives and verbs. In Norwegian, one may also add negative prefixes such as *u-*, *irr-* and *in-*, however, these prefixes cannot be added to all words in this specific selection of words. These words were often not translated automatically and manual correction was therefore performed in consultation with a Norwegian dictionary.

Other factors, e.g., cultural and geographical differences between U.S.A. and Norway, may also influence the translation of analogies. For instance, the *City-in-state* anal-

ogy questions which describe American cities and states, e.g., ⟨*Chicago*:*Illinois*, *Houston*:*Texas*⟩, were deemed less relevant for a Norwegian analogy data set and hence were not included. Instead, we constructed similar analogies consisting of Norwegian cities and counties, e.g., ⟨*Hønefoss*:*Buskerud*, *Stord*:*Hordaland*⟩, where *Hønefoss* and *Stord* are cities and *Buskerud* and *Hordaland* are counties. Like Mikolov et al. (2013a), we constructed a list of 68 Norwegian cities, randomly chosen from Wikipedia's list of Norway's largest cities, and formed about 2,5K questions by connecting every city–county pair to 38 randomly selected other pairs.

After post-processing, the final Norwegian Analogy Test Set comprises 8,944 semantic and 8,863 syntactic questions, i.e., 17,807 analogy questions in total. Table 1 provides an overview of the number of tuples for each of the different relation types, with examples illustrating each of the relation types. The data set is made freely available online.[4]

4 The Norwegian synonymy test set

Another task often used for evaluating distributional semantic models is to automatically determine semantically similar words, or (near-)synonyms, such as *cup* and *mug* or *taxi* and *cab*. Prior to this work there was no freely available, digital synonym resource for Norwegian. In order to create a Norwegian synonym test set we adapted *Norske synonymer blå ordbok*, which is a Norwegian synonym dictionary manually created by Dag Gundersen and published by Kunnskapsforlaget. The dictionary consists of approximately 28,000 headwords and 130,000 reference words. Each headword entry is associated with one or more lexical items, such as synonyms or reference words. Reference words are synonyms that refer to other lexical entries associated with additional synonyms, whereas synonyms do not refer further. We were given access to the dictionary by Kunnskapsforlaget and were allowed to release[5] the data set under a Creative Commons BY-NC-SA 4.0 license. The re-use of this existing resource has several advantages. First, we do not need to use annotators to create a resource from scratch. Instead, we take advantage of an already existing resource with large coverage. Second, the resource is created by professional lexicographers, hence, it should therefore be reliable and of higher quality than a resource created by, say, crowdsourcing. In order to make use of this resource to evaluate word embedding models, the data underwent several steps of processing and refinement: (i) XML parsing, (ii) expansion of spelling variants, (iii) extraction of synonyms. Below we will describe these in more detail.

XML parsing *Norske synonymer blå ordbok* is distributed in XML format and we make use of the XML parser included in the ElementTree library[6] for Python to extract synonym data. For further processing we are interested in the following XML elements: `oppslagsord` (headword), `variant` (spelling variant), `Synonym` (synonym), `henv-ord` (reference word) and `SynonymGruppe` (synonym group). The headwords of interest are associated with a `Kropp` (body) element containing one or more synonym groups, i.e., lists of synonyms grouped by word sense.

[4] `https://github.com/ltgoslo/norwegian-analogies`
[5] `https://github.com/ltgoslo/norwegian-synonyms`
[6] `https://docs.python.org/3.5/library/xml.etree.elementtree.html`

Spelling variants Many Norwegian words have spelling variants, i.e., different forms of the same word. In the XML document, spelling variants are coded either explicitly by the use of `variant` tags or implicitly by the use of certain types of parentheses. For example, the headword *deltaker* 'participant' may also be spelled *deltager*. Spelling variants that are not marked by `variant` tags may be marked with parentheses, e.g., *albu(e)rom* 'elbow-room', *rampet(e)* 'mischievous' or even by nested parentheses, as in *kolonial(vare(r))* 'groceries' and *(land(e))vinning* 'conquest'. These should be expanded with all spelling variants, e.g., *kolonial(vare(r))* expands to *kolonial*, *kolonialvare* and *kolonialvarer*.

We consider spelling variants to be synonyms as they can mutually replace one another. Consequently, we must extend our dictionary by adding spelling variants. Both headwords and synonyms are observed with spelling variants. For each headword with alternate spellings, its synonym list is extended with the variant words. Furthermore, if two (or more) headwords are spelling variants of each other, their synonym lists need to be identical except for the respective variants, thus, we modify their resulting synonym lists to be the union of their initial synonym lists. Some spelling variants are not headwords themselves and such words are added as new entries to the synonym dictionary, associated with the synonym list of the original headword extended with the headword and possible other spelling variants. Finally, wherever a spelling variant occurs in other headwords' synonym lists, these synonym lists are extended with the variants. In contrast to headwords with spelling variants, we only expand synonyms locally, i.e., the synonym *foto(grafi)* 'photography' is replaced by *foto* and *fotografi* only where the synonym explicitly occurs with parentheses.

Synonym extraction Synonyms are initially added to the dictionary without being processed, i.e., the synonym lists include multi-word expressions. However, since the basic units of our semantic vectors will be individual words, the synonym lists need to be further processed. In general, multi-word synonyms are discarded. However, if the entries are of the form *gruble (over)* 'ponder over', *fight (fait)* 'fight' and *ligne (likne)* 'resemble', i.e., with a single word outside of a parenthesis, the synonym entry is replaced by that word. In addition, if the word within the parenthesis is a spelling variant of the added word, the spelling variant is also added to the synonym list.

The resulting synonym dictionary is a list of headwords in which we can look up a particular word and retrieve a list of its synonyms. The synonym lists are simply flat lists, including every synonym of a headword, regardless of their word meaning.

As data-driven distributional methods are based on corpus observations, we finally impose a frequency cut-off on entries to be included in our final data set. We modify the initial synonym dictionary so that each headword, and at least one of its synonyms, must occur 5 or more times in the concatenation of three lemmatized corpora which includes NBDigital[7] and NoWaC[8] in addition to NNC. As long as one synonym occurs 5 or more times, all synonyms of the given headword are retained regardless of their frequencies.

With this cut-off, the resulting dictionary comprises a total of 24,649 headwords associated with 30,756 distinct synonym word types, comprising 106,749 headword–synonym pairs in total. Each headword is associated with 4.33 synonyms on average.

[7]This is a digital text collection from the National Library of Norway (Nasjonalbiblioteket) from which we extracted a total of 13,771 texts in Norwegian Bokmål, comprising approximately 813 million tokens.

[8]Norwegian Web as Corpus; a web-based corpus of Norwegian Bokmål developed at the Department of Linguistics and Scandinavian Studies at the University of Oslo, comprising approx. 687 million tokens.

		Types	
Sentences	Tokens	Full-forms	Lemmas
87,515,520	1,527,414,377	9,016,282	7,886,045

Table 2: Basic corpus counts for our version of the Norwegian Newspaper Corpus

Precision and recall We here describe how a given word vector model is evaluated with respect to our synonym test in terms of task-adapted definitions of precision and recall. The task of the model is to predict synonyms for each headword in the test set, by retrieving the words in the vocabulary which is closest to the target in terms of cosine distance. In a typical classification task one makes a prediction for every instance presented. However, in the task of synonym detection we do not make a prediction for headwords we do not have embeddings for. We here define precision to be the number of headwords for which we predict a correct synonym among its k most similar words, over the number of headwords for which we have embeddings in addition to having embeddings for at least one of its synonyms. Furthermore, we define recall to be the number of headwords for which we predict a correct synonym among its k most similar words, over the total number of headwords in the synonym dictionary. In this way, precision and recall can ultimately be regarded as two variants of accuracy, differing with respect to whether or nor we consider headwords we do not have embeddings for.

5 Training word embeddings for Norwegian

In this section we describe the various word embedding models that we train and later evaluate using the benchmark test sets developed above. We will describe the various word embedding frameworks that we have used, in addition to the Norwegian Newspaper Corpus which we use for training. In order to facilitate replicability and re-use, the word embeddings will be made available online.[9]

The Norwegian Newspaper Corpus Norsk Aviskorpus,[10] or the Norwegian Newspaper Corpus (NNC), contains texts that have been collected from the web edition of 24 Norwegian newspapers since 1998, and the corpus currently contains over 1 billion tokens of Norwegian Bokmål (and 60 million tokens of Norwegian Nynorsk, though not used here), making it the largest corpus of Norwegian. Pre-tokenized texts are available for articles dated 1998–2011, while articles dated 2012–2014 are available as separate XML documents. We have extracted the raw text from the XML documents and performed word tokenization using UDPipe[11] (v.1.1) (Straka et al., 2016), and then combined this with the texts from 1998–2011 (Bokmål only).

We also apply UDPipe to lemmatize[12] the entire tokenized corpus. We used version v.1.2 of the UDPipe toolkit, with a pre-trained model for Norwegian Bokmål trained on the Universal Dependencies (UD) 2.0 version of the Norwegian UD treebanks (Øvrelid and Hohle, 2016; Velldal et al., 2017). Some descriptive statistics for the resulting corpus are summarized in Table 2.

[9] http://vectors.nlpl.eu/repository/

[10] https://www.nb.no/sprakbanken/show?serial=sbr-4&lang=en

[11] http://ufal.mff.cuni.cz/udpipe

[12] Lemmatization refers to the process of morphological analysis to infer the 'base-forms' or 'dictionary-forms' (i.e., lemmas) of 'full-forms' or surface words in running text.

Word embedding frameworks Using the news corpus described above as training data, we will compute word embeddings using three of the most used frameworks; word2vec (Mikolov et al., 2013a), fastText (Bojanowski et al., 2016) and GloVe (Pennington et al., 2014). Moreover, both word2vec and fastText makes available two different estimation strategies; Continuous Bag-of-Words (CBOW) and Skip-gram (SG). Both CBOW and SG embeddings are estimated by training a simple artificial neural network to predict neighboring words, where the word vectors (embeddings) correspond to the hidden layer of the network. The shared intuition is that embeddings that are good at predicting neighboring words are also good at representing word similarity because semantically similar words tend to occur in similar contexts. Hence, the neural network will try to learn embeddings that are maximally similar to the embeddings of their neighboring words and minimally similar to the embeddings of the words which do not occur close by. However, while CBOW learns to predict the target word based on the context words, SG learns to predict the context words given the target word.

Note that fastText is essentially an extension of word2vec that additionally takes into account the internal structure of words, thereby potentially making it more useful for morphologically rich languages. More precisely, fastText embeddings incorporate subword information by representing a word by the sum of the vector representations of its character n-grams (Bojanowski et al., 2016). For the word2vec models we use the re-implementation of the original word2vec tool of Mikolov et al. (2013a) available in the Gensim[13] Python toolkit. The fastText vectors are computed using the open-source library released by Facebook AI Research.[14]

The third framework we will consider is GloVe (Pennington et al., 2014), short for Global Vectors, which attempts to combine explicit count-based models and prediction-based models with local context windows. In the GloVe model, word vectors are trained on the non-zero entries of the global word–word co-occurrence matrix and the training objective is to learn word vector representations so that the dot product between them is equal to the logarithm of the probability of the two words co-occurring (Pennington et al., 2014). The underlying intuition is that the quantitative relation, in terms of the *ratio*, of such co-occurrence probabilities may encode meaning in some form.

Note that, the goal of our evaluation experiments is not to discover the best word embedding model possible, and we consider hyperparameter optimization out-of-scope of this work. We use default values for the hyperparameters of each framework, except for the dimensionality of the GloVe vectors which defaults to 50 but which we here set to 100 – the default for both word2vec and fastText. For each of the five training strategies described above we train separate models on both full-forms and lemmas of NNC.

6 Evaluation experiments

In the experiments reported in this section we will attempt to isolate the effects of text pre-processing (lemmatization vs. full-forms) and choice of word embedding framework, seeking to demonstrate the usefulness of the evaluation resources for ranking the relative performance of different models. We start with the task of analogical reasoning and then move on to synonym detection. Following common practice (Mikolov et al., 2013a), we will report evaluation results restricted to only the 30K most frequent words of the model vocabulary. This restriction involves ignoring analogy questions that includes a word

[13] `https://radimrehurek.com/gensim`

[14] `https://github.com/facebookresearch/fastText`

	Lemmas	Full-forms		
Model	Sem.	Sem.	Syn.	Tot.
Word2Vec CBOW	46.0	38.5	56.6	48.5
Word2Vec SG	56.3	50.3	61.6	56.5
FastText CBOW	56.1	44.1	**70.4**	58.5
FastText SG	**68.1**	**63.5**	67.5	**65.7**
GloVe	59.7	50.9	57.9	54.7

Table 3: Evaluating analogical reasoning: Accuracy on the semantic and syntactic sections and total accuracy on all relation types in the Norwegian Analogy Test Set for the various full-form and lemma embeddings trained on NNC.

not present among the 30K most frequent. For the vocabularies considered here, this in practice means that only roughly half of the analogy questions are considered.

Analogical reasoning

In this section we compare the performance of word2vec (CBOW + SG), fastText (CBOW + SG) and GloVe embeddings, trained on both full-form and lemmatized versions of NNC, computing accuracies on both the semantic and syntactic analogy questions. The results are presented in Table 3.

In general, we observe that the SG embeddings tend to yield better accuracies than the corresponding CBOW embeddings; this holds for both word2vec and fastText. Moreover, we see that the lemma embeddings consistently yield much higher accuracies than the corresponding full-form embeddings; this holds across all five frameworks. This makes sense, as these models can take advantage of the additional data provided by the lemmatization: the normalization means there will be fewer unique word types in the corpus, and hence potentially more distributional information per word in the vocabulary.

In terms of absolute accuracies we observe that fastText embeddings generally predict the analogies better than both word2vec and GloVe. In particular, the fastText SG embeddings yield the best semantic accuracy, for both the lemma (Acc=68.1%) and full-form (Acc=63.5%) configurations, and also the best total accuracy (65.7%). For the syntactic subset, however, fastText CBOW yields the best results.

As discussed above, fastText embeddings take into account the internal structure of words (a given word embedding is composed by the character n-gram embeddings). This might be particularly beneficial for embeddings trained for Norwegian, due to the way the language handles *compounds*. For instance, the nominal phrase 'table tennis' is written as a single word *bordtennis* in Norwegian. In particular, we can reasonably expect that subword information might be beneficial when predicting syntactic analogies, as these are related to the morphology of the words. It is worth noting that fastText is also superior to the other word embedding models in terms of training time. For example, training fastText SG on the full-form version of NNC takes about 2 hours, as opposed to more than twice that for the word2vec SG model.

Methodological concerns There are some methodological issues that should be taken into account regarding the analogy evaluations. First, as seen in Table 1, the different relation types contain an unbalanced number of analogy questions. For instance, the *All capital cities* relation type contains a total of 4,524 questions, accounting for a little

	k=1		*k*=5		*k*=10	
Model	P	R	P	R	P	R
Word2Vec CBOW	**10.3**	**8.8**	**21.3**	**18.2**	**26.5**	**22.7**
Word2Vec SG	10.1	8.6	20.8	17.8	25.9	22.2
FastText CBOW	6.9	5.9	16.9	14.5	22.1	18.9
FastText SG	8.6	7.4	19.9	17.0	25.1	21.5
GloVe	8.4	7.2	18.8	16.1	23.7	20.3

Table 4: Evaluating synonym detection: Precision and recall among the 1, 5, and 10 most similar words, computed for word2vec CBOW and SG, fastText CBOW and SG and GloVe lemma embeddings trained on NNC (only considering the 30K most frequent words in the vocabulary at test time).

over 50% of the semantic questions. Second, one might argue that the relation type *Nationality adjective* which is specified as a syntactic relation in the test set, might as well be categorized as semantic, as also pointed out by Fares et al. (2017). In the context of our evaluation results, this intuition is supported by observing the performance of the lemma models on this relation. As previously stated, it is not meaningful to evaluate lemma embeddings on (most of the) syntactic analogies, given that they often rely on morphological information not present in the normalized lemma model, but this is not the case for relation types like *Nationality adjective*. For example, the word2vec SG lemma embeddings trained on NNC yield an accuracy of 97.2% for this category. Nevertheless, we follow the initial category sectioning as proposed by Mikolov et al. (2013a).

Synonym detection

We now turn to evaluation of the various word embedding models for the task of synonym detection, based on our synonym test set described in Section 4. Just as in the case of the semantic analogies, we here only consider lemma embeddings, as the synonym dictionary entry words are lemmas. For all models we report precision and recall among the 1, 5, and 10 nearest neighbors of the target words. The results are shown in Table 4.

We see that the word2vec CBOW model yields the overall best results in terms of both precision and recall across all values of *k*. Moreover, fastText CBOW is generally the worst performing model across the board. In general, the relative differences in precision and recall scores between the various word embedding frameworks are consistent across different values of *k*. Regardless of relative differences, we find that both precision and recall scores across word embedding frameworks are quite low. This is in accordance with other studies on the use of word embeddings for synonym extraction for English, for example the study performed by Leeuwenberg et al. (2016), reflecting the fact that the evaluation criterion is rather strict. To shed more light on the performance, we performed a manual error analysis of the most similar words found by the best performing model in terms of precision, namely word2vec CBOW.

Manual error analysis We randomly choose a selection of 50 words for which none of the predicted synonyms were considered correct in the automatic evaluation. Based on the error categories proposed by Leeuwenberg et al. (2016), we manually categorize the 1st and 2nd most similar words found for each chosen word, also showing the error counts of each. The results of the analysis, along with some examples of each category, are given

Category	# 1st	# 2nd	Example
Human-judged synonyms	6	2	fjomp / dust, styrkeprøve / kraftprøve
Spelling variants	6	2	blackout / black-out, idérikdom / iderikdom
Related	11	17	innbilning / vrøvl, nervøsitet / pessimisme
Unrelated/unknown	11	15	ærend / skjulested, intendere / uhistorisk
Names	2	1	avtrede / Kanofarten, ponere / Zillertal
Co-hyponyms	2	4	kobra / tarantell, styrkeprøve / manndomsprøve
Inflections	4	3	bløt / bløte, samstemme / samstemt
Hyponyms	6	3	bue / fiolinbue, utslipp / CO2-utslipp
Contrastive	2	-	ekvivalent / motstykke, negativ / positiv
Hypernym	-	1	amfora / leirkrukke
Foreign	1	2	futhark / fæstkultur, futhark / Cuius

Table 5: Manual categorization of errors for 50 randomly selected target words for which none of the candidate synonyms detected by the word2vec CBOW lemma embeddings were considered correct according to the test set. We categorize both the 1st and 2nd closest neighbor for each target.

in Table 5. The analysis shows that only 11 of the 1st most similar and 15 of the 2nd most similar words are deemed completely unrelated to the headword. The rest are largely words that are semantically related (e.g., *nervøsitet/pessisme* 'nervousness/pessimism') or otherwise in another type of semantic relation, e.g., co-hyponymy or hyponymy, or reflecting information that is not in the dictionary, such as additional synonymy relations (e.g., *fjomp/dust* 'fool/idiot') or spelling variants (e.g., *blackout/black-out*).

7 Conclusion

In this article we have presented two benchmark data sets that enable intrinsic evaluation of distributional semantic models for Norwegian. The data sets target the tasks of analogical reasoning and synonym detection. While such resources are available for English, they did not exist for Norwegian prior to this work. Furthermore, we have produced large-coverage semantic vectors trained using several word embedding frameworks. Finally, we have demonstrated the usefulness of the evaluation resources created in the context of this paper for ranking the relative performance of different word embedding models. All resources are made freely available.

References

Agirre, E., Alfonseca, E., Hall, K., Krabalova, J., Pasca, M., and Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 19–27, Boulder, CO, USA.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Fares, M., Kutuzov, A., Oepen, S., and Velldal, E. (2017). Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In

Proceedings of the 21st Nordic Conference of Computational Linguistics, pages 271–276, Gothenburg, Sweden.

Finkelstein, L., Gabrilovich, E., Mathias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Jurgens, D. A., Turney, P. D., Mohammad, S. M., and Holyoak, K. J. (2012). SemEval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the 1st Joint Conference on Lexical and Computational Semantics*, page 356–364, Montreal, Canada.

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

Leeuwenberg, A., Velab, M., Dehdaribc, J., and van Genabith, J. (2016). A minimally supervised approach for synonym extraction with word embeddings. *The Prague Bulletin of Mathematical Linguistics*, 105:111–142.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013b). 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, GA, USA.

Øvrelid, L. and Hohle, P. (2016). Norwegian Universal Dependencies. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1579–1585, Portorož, Slovenia.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, page 1532–1543, Doha, Qatar.

Stadsnes, C. (2018). Evaluating semantic vectors for norwegian. Master's thesis, University of Oslo.

Straka, M., Hajič, J., and Straková, J. (2016). Udpipe: Trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, Portorož, Slovenia.

Velldal, E., Øvrelid, L., and Hohle, P. (2017). Joint UD parsing of Norwegian Bokmål and Nynorsk. In *Proceedings of the 11th Nordic Conference of Computational Linguistics*, pages 1–10, Gothenburg, Sweden.