The Interplay of Semantics and Morphology in Word Embeddings

Oded Avraham and **Yoav Goldberg**
Computer Science Department
Bar-Ilan University
Ramat-Gan, Israel
{oavraham1,yoav.goldberg}@gmail.com

Abstract

We explore the ability of word embeddings to capture both semantic and morphological similarity, as affected by the different types of linguistic properties (surface form, lemma, morphological tag) used to compose the representation of each word. We train several models, where each uses a different subset of these properties to compose its representations. By evaluating the models on semantic and morphological measures, we reveal some useful insights on the relationship between semantics and morphology.

1 Introduction

Word embedding models learn a space of continuous word representations, in which similar words are expected to be close to each other. Traditionally, the term *similar* refers to *semantic* similarity (e.g. *walking* should be close to *hiking*, and *happiness* to *joy*), hence the model performance is usually evaluated using semantic similarity datasets. Recently, several works introduced morphology-driven models motivated by the poor performance of traditional models on morphologically complex words. Such words are often rare, and there is not enough evidence to model them correctly. The morphology-driven models allow pooling evidence from different words which have the same base form. These models work by learning per-morpheme representations rather than just per-word ones, and compose the representing vector of each word from those of its morphemes – as derived from a supervised or unsupervised morphological analysis – and (optionally) its surface form (e.g. *walking* = $f(v_{walk}, v_{ing}, v_{walking})$).

The works differ in the way they acquire morphological knowledge (from using linguistically derived morphological analyzers on one end, to approximating morphology using substrings while relying on the concatenative nature of morphology, on the other) and in the model form (cDSMs (Lazaridou et al., 2013), RNN (Luong et al., 2013), LBL (Botha and Blunsom, 2014), CBOW (Qiu et al., 2014), SkipGram (Soricut and Och, 2015; Bojanowski et al., 2016), GGM (Cotterell et al., 2016)). But essentially, they all show that breaking a word into morphological components (base form, affixes and potentially also the complete surface form), learning a vector for each component, and representing a word as a composition of these vectors improves the models semantic performance, especially on rare words.

In this work we argue that these models capture two distinct aspects of word similarity, *semantic* (e.g. *sim(walking, hiking)* > *sim(walking, eating)*) and *morphological* (e.g. *sim(walking, hiking)* > *sim(walking, hiked)*), and that these two aspects are at odds with each other (should *sim(walking, hiking)* be lower or higher than *sim(walking, walked)*?). The *base form* component of the compositional models is mostly responsible for semantic aspects of the similarity, while the *affixes* are mostly responsible for morphological similarity.

This analysis brings about several natural questions: is the combination of semantic and morphological components used in previous work ideal for every purpose? For example, if we exclude the morphological component from the representations, wouldn't it improve the semantic performance? What is the contribution of using the surface form? And do the models behave differently on common and rare words? We explore these questions in order to help the users of morphology-driven models choose the right configuration for their needs: semantic or morphological performance, on common or rare words.

Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 422–426,
Valencia, Spain, April 3-7, 2017.

We compare different configurations of morphology-driven models, while controlling for the components composing the representation. We then separately evaluate the semantic and morphological performance of each model, on rare and on common words. We focus on *inflectional* (rather than *derivational*) morphology. This is due to the fact that derivations (e.g. *affected* → *unaffected*) often drastically change the meaning of the word, and therefore the benefit of having similar representations for words with the same derivational base is questionable, as discussed by Lazaridou et al (2013) and Luong et al (2013). Inflections (e.g. *walked* → *walking*), in contrast, preserve the word lexical meaning, and only change its grammatical categories values.

Our experiments are performed on Modern Hebrew, a language with rich inflectional morphological system. We build on a recently introduced evaluation dataset for semantic similarity in Modern Hebrew (Avraham and Goldberg, 2016), which we further extend with a collection of rare words. We also create datasets for morphological similarity, for common and rare words. Hebrew's morphology is not concatenative, so unlike most previous work we do not break the words into base and affixes, but instead rely on a morphological analyzer and represent words using their *lemmas* (corresponding to the base form) and their *morphological tags* (from which the morphological forms are derived, corresponding to affixes). This allow us to have a finer grained control over the composition, separating inflectional from derivational processes. We also compare to a strong character ngram based model, that mixes the different components and does not allow finer-grained distinctions.

We observe a clear trade-off between the morphological and semantic performance – models that excel on one metric perform badly on the other. We present the strengths and weaknesses of the different configurations, to help the users choose the one that best fits their needs. To the best of our knowledge, this work is the first to make a comprehensive comparison between various configurations of morphology-driven models,[1] as well as the first to evaluate both semantic and morphological performance of such models. While our experiments focus on Modern Hebrew due to the availability of a reliable semantic similarity dataset, we believe our conclusions hold more generally.

2 Models

Our model form is a generalization of the fastText model (Bojanowski et al., 2016), which in turn extends the skip-gram model of Mikolov et al (2013). The skip-gram model takes a sequence of words $w_1, ..., w_T$ and a function s assigning scores to (word, context) pairs, and maximizes

$$\sum_{t=1}^{T}\left(\sum_{w_c\in\mathcal{C}_t}\ell(s(w_t,w_c))+\sum_{w'_c\in\mathcal{N}_t}\ell(-s(w_t,w'_c))\right)$$

where ℓ is the log-sigmoid loss function, $\mathcal{C}_t$ is a set of context words, and $\mathcal{N}_t$ is a set of negative examples sampled from the vocabulary. $s(w_t, w_c)$ is defined as $s(w_t, w_c) = \mathbf{u}_{w_t}^{\top}\mathbf{v}_{w_c}$ (where $\mathbf{u}_{w_t}$ and $\mathbf{v}_{w_c}$ are the embeddings of the focus and the context words).

Bojanowski et al (2016) replace the word representation $\mathbf{v}_{w_t}$ with the set of character ngrams appearing in it: $\mathbf{v}_{w_c} = \sum_{g\in\mathcal{G}(w_t)}\mathbf{v}_g$ where $\mathcal{G}(w_t)$ is the set of n-grams appearing in w_t. The n-grams are used to approximate the morphemes in the target word.

We generalize Bojanowski et al (2016) by replacing the set of ngrams $\mathcal{G}(w)$ with a set $\mathcal{P}(w)$ of explicit linguistic properties. Each word w_t is then composed as the sum of the vectors of its linguistic properties: $\mathbf{v}_{w_t} = \sum_{p\in\mathcal{P}(w_t)}\mathbf{v}_p$. The linguistic properties we consider are the surface form of the word (W), it's lemma (L) and its morphological tag (M)[2]. The lemma corresponds to the base-form, and the morphological tag encodes the grammatical properties of the word, from which its inflectional affixes are derived (a similar approach was taken by Cotterell and Schütze (2015)). Moving from a set of n-grams to a set of explicit linguistic properties, allows finer control of the kinds of information in

[1] Among the previous work mentioned above, only few explored configurations other than (base + affixes) or (surface + base + affixes). Lazaridou et al (2013) and Luong et al (2013) trained models which represent a word by its base only, and showed that these models performs worse than the compositional ones (base + affixes). However, the poor results for the base-only models were mainly attributed to undesirable capturing of derivational similarity, e.g. *(affected, unaffected)*. Working with a more linguistically informed morphological analyzer allows us to tease apart inflectional from derivational processes, leading to different results.

[2] The lemma and morphological tag for a word in context are obtained using a morphological analyzer and disambiguator. Then, each value of lemma/tag/surface from is associated with a trainable embedding vector.

the word representation. We train models with different subsets of $\{W, L, M\}$.

3 Experiments and Results

Our implementation is based on the *fastText*[3] library (Bojanowski et al., 2016), which we modify as described above. We train the models on the Hebrew Wikipedia (∼4M sentences), using a window size of 2 to each side of the focus word, and dimensionality of 200. We use the morphological disambiguator of Adler (2007) to assign words with their morphological tags, and the inflection dictionary of *MILA* (Itai and Wintner, 2008) to find their lemmas. For example, for the words נסתכל (*[we will] look [at]*), הסתכלה (*[she] looked [at]*) and הסתכל (*[he] looked [at]*) are assigned the tags *VB.MF.P.1.FUTURE*, *VB.F.S.3.PAST* and *VB.M.S.3.PAST* respectively, and share the lemma הסתכל. We train the models for the subsets $\{W\}$, $\{L\}$, $\{W, L\}$, $\{W, M\}$ and $\{W, L, M\}$, as well as the original fastText (n-grams) model. Finally, we evaluate each model on several datasets, using both semantic and morphological performance measures.[4]

Semantic Evaluation Measure The common datasets for semantic similarity[5] have some notable shortcomings as noted in (Avraham and Goldberg, 2016; Faruqui et al., 2016; Batchkarov et al., 2016; Linzen, 2016). We use the evaluation method (and corresponding Hebrew similarity dataset) that we have introduced in a previous work (Avraham and Goldberg, 2016) (AG). The AG method defines an annotation task which is more natural for human judges, resulting in datasets with improved annotator-agreement scores. Furthermore, the AG's evaluation metric takes annotator agreement into account, by putting less weight on similarities that have lower annotator agreement.

An AG dataset is a collection of target-groups, where each group contains a target word (e.g. *singer*) and three types of candidate words: *positives* which are words "similar" to the target (e.g. *musician*), *distractors* which are words "related but dissimilar" to the target (e.g. *microphone*), and *randoms* which are not related to the target at all (e.g *laptop*). The human annotators are asked to rank the positive words by their similarity to the target word (distractor and random words are not annotated by humans and are automatically ranked below the positive words). This results in a set of triples of a target word w and two candidate words c_1, c_2, coupled with a value indicating the confidence of ranking $sim(w, c_1) > sim(w, c_2)$ by the annotators. A model is then scored based on its ability to correctly rank each triple, giving more weight to highly-confident triples. The scores range between 0 (all wrong answers) to 1 (perfect match with human annotators).

We use this method on two datasets: the AG dataset from (Avraham and Goldberg, 2016) (*SemanticSim*, containing 1819 triples), and a new dataset we created in order to evaluate the models on rare words (similar to RW (Luong et al., 2013)). The rare-words dataset (*SemanticSim-Rare*) follows the structure of *SemanticSim*, but includes only target words that occur less than 100 times in the corpus. It contains a total of 163 triples, all of the type positive vs. random (we find that for rare words, distinguishing similar words from random ones is a hard enough task for the models).

Morphological Evaluation Measure Cotterrel and Schütze (2015) introduced the MorphoDist$_k$ measure, which quantifies the amount of morphological difference between a target word and a list of its k most similar words. We modify MorphDist$_k$ measure to derive MorphSim$_k$, a measure that ranges between 0 and 1, where 1 indicates total morphological compatibility. The MorphDist measure is defined as: $MorphoDist_k(w) = \sum_{w' \in \mathcal{K}_w} \min_{m_w, m_{w'}} d_h(m_w, m_{w'})$ where $\mathcal{K}_w$ is the set of top-k similarities of w, m_w and $m_{w'}$ are possible morphological tags of w and w' respectively (there may be more than one possible morphological interpretation per word), and d_h is the Hamming distance between the morphological tags. *MorphoDist* counts the *total number* of incompatible morphological components. MorphSim$_k$ calculates the *average rate* of *compatible* morphological values. More formally, $MorphoSim_k(w) = 1 - \frac{MorphoDist_k(w)}{k \cdot |m_w|}$, where $|m_w|$ is the number of grammatical components specified in w's morphological tag.

We use k=10 and calculate the average *MorphoSim* score over 100 randomly chosen words.

[3] https://github.com/facebookresearch/fastText

[4] Our code is available on `https://github.com/oavraham1/prop2vec`, our datasets on `https://github.com/oavraham1/ag-evaluation`

[5] E.g., WordSim353 (Finkelstein et al., 2001), RW (Luong et al., 2013) and SimLex999 (Hill et al., 2015)

	1st	2nd	3rd
W	הביטה:gaze:VB.F.S.3.PAST	חייכה:smile:VB.F.S.3.PAST	מתייפחת:cry:VB.F.S.3.PRESENT
L	הביטי:gaze:VB.F.S.2.IMPERATIVE	התבונן:watch:VB.M.S.3.PAST	בהו:stare:VB.MF.P.3.PAST
WL	נביט:gaze:VB.MF.P.1.FUTURE	התבוננה:watch:VB.F.S.3.PAST	בוהה:stare:VB.F.S.3.PRESENT
WM	חייכה:smile:VB.F.S.3.PAST	נחבלה:injure:VB.F.S.3.PAST	נשפה:blow:VB.F.S.3.PAST
LM	הביטה:gaze:VB.F.S.3.PAST	התבוננה:watch:VB.F.S.3.PAST	זזה:move:VB.F.S.3.PAST
WLM	הביטה:gaze:VB.F.S.3.PAST	התבוננה:watch:VB.F.S.3.PAST	פסעה:walk:VB.F.S.3.PAST

Table 1: Top-3 similarities for the word הסתכלה (*[she] looked [at]*).
Each entry is of the form *[word:lexical meaning:morphological tag]*. Green-colored items share the semantic/inflection of the target word, while red-colored indicate a divergence. In the morphological tags: M/F/MF indicate masculine/feminine/both, P/S indicate plural/singular, 1/2/3 indicate 1st/2nd/3rd person.

To evaluate the morphological performance on rare words, we run another benchmark (*MorphoSimRare*) in which we calculate the average *MorphoSim* score over the 35 target words of the *SemanticSimRare* dataset.

Qualitative Results To get an impression of the differences in behavior between the models, we queried each model for the top similarities of several words (calculated by cosine similarity between words vectors), focusing on rare words. Table 1 presents the top-3 similarities for the word הסתכלה (*[she] looked [at]*), which occurs 17 times in the corpus, under the different models. Unsurprisingly, the lemma component has a positive effect on semantics, while the tag component improves the morphological performance. It also shows a clear trade-off between the two aspects – as models which perform the best on semantics are the worst on morphology. This behavior is representative of the dozens of words we examined.

Quantitative Results We compare the different models on the different measures, and also compare to the state-of-the-art n-gram based fastText model of Bojanowski et al (2016) that does not require morphological analysis. The results (Table 2) highlight the following:

1. There is a trade-off between semantic and morphological performance – improving one aspect comes at the expense of the other: the lemma component improves semantics but hurts morphology, while the opposite is true for the tag component. The common practice of using both components together is a kind of compromise: the *LM*, *WLM* and *n-grams* models are not the best nor the worst on any measure.

2. The impacts of the lemma and the tag components are much larger when dealing with rare words: comparing to *W*, *WL* is only 1.7% better on *SS* and 3.8% worse on *MS*, while it's 16.3% better and 11.9% worse on *SSR* and *MSR* (respectively). Similarly, *WM* is only 2.8% worse than *W* on *SS* and 44.9% better on *MS*, while it's 21.8% worse and 75.7% better on *SSR* and *MSR* (respectively).

	SS	*SSR*	*MS*	*MSR*
W	0.707	0.675	0.626	0.569
L	0.713	**0.816**	**0.491**	**0.339**
WL	**0.719**	0.785	0.602	0.501
WM	**0.687**	**0.528**	**0.907**	**1**
LM	0.707	0.693	0.887	0.996
WLM	0.716	0.748	0.882	**1**
n-grams	0.712	0.767	0.71	0.866

Table 2: Results on *SemanticSim (SS)*, *SemanticSimRare (SSR)*, *MorphoSim (MS)* and *MorphoSimRare (MSR)*. The best result for each measure is green, the worst is red.

3. Simply lemmatizing the words is very effective for capturing semantic similarity. This is especially true for the rare words, in which the L model clearly outperform all others. For the common words, we see a small drop compared to including the surface form as well (*WL*, *WLM*). This is attributed to cases in which some of the semantics lies within the word's morphological template, for example: in *W* model, most similar words for the masculine verb נפל (*fell*) are associated with *a soldier* (which is a masculine noun): נהרג (was killed), נפגע (was injured), while the similarities of the feminine form נפלה are associated with *a land* or *a state* (both are feminine nouns): סופחה (was annexed), נכבשה (was occupied). In *L* model – נפלה and נפל share a single, less accurate representation (somewhat similarly to representations of ambiguous words). This suggests using different compositions for common and rare words.

4 Conclusions

Our key message is that users of morphology-driven models should consider the trade-off between the different components of their representations. Since the goal of most works on morphology-driven models was to improve *semantic* similarity, the configurations they used (which combine both semantic and morphological components) were probably not the best choices: we show that using the lemma component (either alone or together with the surface form) is better. Indeed, excluding the morphological component will make the morphological similarity drop, but it's not necessarily a problem for every task. One should include the morphological component in the embeddings only for tasks in which morphological similarity is required and cannot be handled by other means. A future work can be to perform an extrinsic evaluation of the different models in various downstream applications. This may reveal which kinds of tasks benefit from morphological information, and which can be done better by a pure semantic model.

Acknowledgements

The work was supported by the Israeli Science Foundation (grant number 1555/15).

References

Menahem Meni Adler. 2007. *Hebrew morphological disambiguation: An unsupervised stochastic word-based approach*. Ph.D. thesis, Ben-Gurion University of the Negev.

Oded Avraham and Yoav Goldberg. 2016. Improving reliability of word similarity evaluation by redesigning annotation task and performance measure. *ACL Workshop on Evaluating Vector Space Representations for NLP*, page 106.

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. *ACL Workshop on Evaluating Vector Space Representations for NLP*, page 7.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Jan A Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. In *ICML*, pages 1899–1907.

Ryan Cotterell and Hinrich Schütze. 2015. Morphological word-embeddings. In *Proc. of NAACL*.

Ryan Cotterell, Hinrich Schütze, and Jason Eisner. 2016. Morphological smoothing and extrapolation of word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1651–1660.

Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi, and Chris Dyer. 2016. Problems with evaluation of word embeddings using word similarity tasks. *ACL Workshop on Evaluating Vector Space Representations for NLP*, page 30.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4).

Alon Itai and Shuly Wintner. 2008. Language resources for Hebrew. *Language Resources and Evaluation*, 42(1):75–98, March.

Angeliki Lazaridou, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2013. Compositional-ly derived representations of morphologically complex words in distributional semantics. In *ACL (1)*, pages 1517–1526. Citeseer.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. *ACL Workshop on Evaluating Vector Space Representations for NLP*, page 13.

Thang Luong, Richard Socher, and Christopher D Manning. 2013. Better word representations with recursive neural networks for morphology. In *CoNLL*, pages 104–113.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Siyu Qiu, Qing Cui, Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Co-learning of word representations and morpheme representations. In *COLING*, pages 141–150.

Radu Soricut and Franz Och. 2015. Unsupervised morphology induction using word embeddings. In *Proc. NAACL*.