Morph-fitting: Fine-Tuning Word Vector Spaces with Simple Language-Specific Rules

Ivan Vulić1, Nikola Mrkšić1, Roi Reichart2
Diarmuid Ó Séaghdha3, Steve Young1, Anna Korhonen1
1 University of Cambridge 2 Technion, Israel Institute of Technology 3 Apple Inc.
{iv250,nm480,sjy11,alk23}@cam.ac.uk doseaghdha@apple.com roiri@ie.technion.ac.il

Abstract
Morphologically rich languages accentuate two properties of distributional vector space models: 1) the difficulty of inducing accurate representations for low-frequency word forms; and 2) insensitivity to distinct lexical relations that have similar distributional signatures. These effects are detrimental for language understanding systems, which may infer that inexpensive is a rephrasing for expensive or may not associate acquire with acquires. In this work, we propose a novel morph-fitting procedure which moves past the use of curated semantic lexicons for improving distributional vector spaces. Instead, our method injects morphological constraints generated using simple language-specific rules, pulling inflectional forms of the same word close together and pushing derivational antonyms far apart. In intrinsic evaluation over four languages, we show that our approach:
1) improves low-frequency word estimates; and
2) boosts the semantic quality of the entire word vector collection. Finally, we show that morph-fitted vectors yield large gains in the downstream task of dialogue state tracking, highlighting the importance of morphology for tackling long-tail phenomena in language understanding tasks.

1 Introduction
Word representation learning has become a research area of central importance in natural language processing (NLP), with its usefulness demonstrated across many application areas such as parsing (Chen and Manning, 2014; Johannsen et al., 2015), machine translation (Zou et al., 2013), and many others (Turian et al., 2010; Collobert et al., 2011). Most prominent word representation techniques are grounded in the distributional hypothesis (Harris, 1954), relying on word co-occurrence information in large textual corpora (Curran, 2004; Turney and Pantel, 2010; Mikolov et al., 2013; Mnih and Kavukuçoglu, 2013; Levy and Goldberg, 2014; Schwartz et al., 2015, i.a.).

Morphologically rich languages, in which “substantial grammatical information...is expressed at word level” (Tsarfaty et al., 2010), pose specific challenges for NLP. This is not always considered when techniques are evaluated on languages such as English or Chinese, which do not have rich morphology. In the case of distributional vector space models, morphological complexity brings two challenges to the fore:

1. Estimating Rare Words: A single lemma can have many different surface realisations. Naively treating each realisation as a separate word leads to sparsity problems and a failure to exploit their shared semantics. On the other hand, lemmatising the entire corpus can obfuscate the differences that exist between different word forms even though they share some aspects of meaning.

2. Embedded Semantics: Morphology can encode semantic relations such as antonymy (e.g. literate and illiterate, expensive and inexpensive) or (near-)synonymy (north, northern, northerly).

In this work, we tackle the two challenges jointly by introducing a resource-light vector space fine-tuning procedure termed morph-fitting. The proposed method does not require curated knowledge bases or gold lexicons. Instead, it makes use of the observation that morphology implicitly encodes semantic signals pertaining to synonymy (e.g., German word inflections katalanisch, katalanischer denote the same semantic concept in different grammatical roles), and antonymy (e.g., mature vs. immature), capitalising on the
proliferation of word forms in morphologically rich languages. Formalised as an instance of the post-processing semantic specialisation paradigm (Faruqui et al., 2015; Mrkšić et al., 2016), morph-fitting is steered by a set of linguistic constraints derived from simple language-specific rules which describe (a subset of) morphological processes in a language. The constraints emphasise similarity on one side (e.g., by extracting morphological synonyms), and antonymy on the other (by extracting morphological antonyms), see Fig. 1 and Tab. 2.

The key idea of the fine-tuning process is to pull synonymous examples described by the constraints closer together in the transformed vector space, while at the same time pushing antonymous examples away from each other. The explicit post-hoc injection of morphological constraints enables: a) the estimation of more accurate vectors for low-frequency words which are linked to their high-frequency forms by the constructed constraints;¹ this tackles the data sparsity problem; and b) specialising the distributional space to distinguish between similarity and relatedness (Kiela et al., 2015), thus supporting language understanding applications such as dialogue state tracking (DST).²

As a post-processor, morph-fitting allows the integration of morphological rules with any distributional vector space in any language: it treats an input distributional word vector space as a black box and fine-tunes it so that the transformed space reflects the knowledge coded in the input morphological constraints (e.g., Italian words rispettoso and irrispettoso should be far apart in the transformed vector space, see Fig. 1). Tab. 1 illustrates the effects of morph-fitting by qualitative examples in three languages: the vast majority of nearest neighbours are “morphological” synonyms.

We demonstrate the efficacy of morph-fitting in four languages (English, German, Italian, Russian), yielding large and consistent improvements on benchmarking word similarity evaluation sets such as SimLex-999 (Hill et al., 2015), its multilingual extension (Leviant and Reichart, 2015), and SimVerb-3500 (Gerz et al., 2016). The improvements are reported for all four languages, and with a variety of input distributional spaces, verifying the robustness of the approach.

We then show that incorporating morph-fitted vectors into a state-of-the-art neural-network DST model results in improved tracking performance, especially for morphologically rich languages. We report an improvement of 4% on Italian, and 6% on German when using morph-fitted vectors instead of the distributional ones, setting a new state-of-the-art DST performance for the two datasets.³

³There are no readily available DST datasets for Russian.

---

Table 1: The nearest neighbours of three example words (expensive, slow and book) in English, German and Italian before (top) and after (bottom) morph-fitting.

| en_expensive | de_teure | it_costoso | en_slow | de_langsam | it_lento | en_book | de_buch | it_libro |
|-------------|---------|-----------|---------|-----------|---------|---------|---------|---------|
| costly      | teuren  | dispensoso| fast    | almahlich | lenitissimo| books   | sachbuch | romanzo |
| costlier    | kostspielig | remunerativo | slow    | rasch     | lenta    | memoir  | buchs   | racconto |
| cheaper     | aufwändige | kostenintensive | slowed  | gemächlich | inesorabile | novel   | romandebit | volumetto |
| prohibitively | preis | | slower  | schnell   | graduale  | storybooks | büchlein | saggio |
| pricey      | aufwendige | | slowing | explosionsartig | | bluhr | pamphlet | ecclesiastische |

| expensiveness | teures | costosa | slowing | langsamer | lenit | booked | bücher | libri |
|------------|--------|--------|---------|-----------|------|--------|--------|-------|
| costly     | teurem | costose | slowed  | langsames | lente | rebook | buch | libra |
| ruinously  | teurer | dispensoso | slows   | langsamen | veloce | rebooked | büches | libre |

---

¹For instance, the vector for the word *katalanischem* which occurs only 9 times in the German Wikipedia will be pulled closer to the more reliable vectors for *katalanisch* and *katalanischer*, with frequencies of 2097 and 1383 respectively.

²Representation models that do not distinguish between synonyms and antonyms may have grave implications in downstream language understanding applications such as spoken dialogue systems: a user looking for ‘an affordable Chinese restaurant in west Cambridge’ does not want a recommendation for ‘an expensive Thai place in east Oxford’.

³There are no readily available DST datasets for Russian.

---

Figure 1: Morph-fitting in Italian. Representations for *rispettoso*, *rispettosa*, *rispettosi* (EN: respectful), are pulled closer together in the vector space (solid lines; ATTRACT constraints). At the same time, the model pushes them away from their antonyms (dashed lines; REPEL constraints) *irrispettoso*, *irrispettosa*, *irrispettosi* (EN: disrespectful), obtained through morphological affix transformation captured by language-specific rules (e.g., adding the prefix *ir*- typically negates the base word in Italian)
where word frequency over 10, see Tab. 3. We then extract (Sect. 2.2) are used as input for the vector space

The A

A

exemplifying the simplicity and portability of the approach.

DE

for them) without any linguistic training. What is more, the rules morphological rules (or at least with a reasonable subset of them) are not driven by the input A

S

constraints (e.g.

similarity

antonymy constraints (e.g.

nimble and clumsy)

sets of linguistic constraints from these (large) vocabularies using a set of simple language-specific if-then-else rules, see Tab. 2.4 These constraints (Sect. 2.2) are used as input for the vector space post-processing ATTRACT-REPEL algorithm (outlined in Sect. 2.1).

2.1 The ATTRACT-REPEL Model

The ATTRACT-REPEL model, proposed by Mrkšić et al. (2017b), is an extension of the PARAGRAM procedure proposed by Wieting et al. (2015). It provides a generic framework for incorporating similarity (e.g. successful and accomplished) and antonymy constraints (e.g. nimble and clumsy) into pre-trained word vectors. Given the initial vector space and collections of ATTRACT and REPEL constraints A and R, the model gradually modifies the space to bring the designated word vectors closer together or further apart. The method’s cost function consists of three terms. The first term pulls the ATTRACT examples \((x_l, x_r) \in A\) closer together. If \(B_A\), denotes the current mini-batch of ATTRACT examples, this term can be expressed as:

\[
A(B_A) = \sum_{(x_l, x_r) \in B_A} \left(ReLU \left(\delta_{att} + x_l t_i - x_r x_i\right) + ReLU \left(\delta_{att} + x_r t_i - x_l x_i\right)\right)
\]

where \(\delta_{att}\) is the similarity margin which determines how much closer synonymous vectors should be to each other than to each of their respective negative examples. \(ReLU(x) = \max(0, x)\) is the standard rectified linear unit (Nair and Hinton, 2010). The ‘negative’ example \(t_i\) for each word \(x_i\) in any ATTRACT pair is the word vector closest to \(x_i\) among the examples in the current mini-batch (distinct from its target synonym and itself). This means that this term forces synonymous words from the in-batch ATTRACT constraints to be closer to one another than to any other word in the current mini-batch.

The second term pushes antonyms away from each other. If \((x_l, x_r) \in B_R\) is the current mini-batch of REPEL constraints, this term can be expressed as follows:

\[
R(B_R) = \sum_{(x_l, x_r) \in B_R} \left(ReLU \left(\delta_{reg} + x_l x_r - x_r t_i\right) + ReLU \left(\delta_{reg} + x_r x_l - x_l t_i\right)\right)
\]

In this case, each word’s ‘negative’ example is the (in-batch) word vector furthest away from it (and distinct from the word’s target antonym). The intuition is that we want antonymous words from the input REPEL constraints to be further away from each other than from any other word in the current mini-batch; \(\delta_{reg}\) is now the repel margin.

The final term of the cost function serves to retain the abundance of semantic information encoded in the starting distributional space. If \(x_i^{init}\) is the initial distributional vector and \(V(B)\) is the set of all vectors present in the given mini-batch, this term (per mini-batch) is expressed as follows:

\[
R(B_A, B_R) = \sum_{x_i \in V(B_A) \cup B_R} \lambda_{reg} \left\|x_i^{init} - x_i\right\|_2^2
\]

where \(\lambda_{reg}\) is the L2 regularisation constant. This term effectively pulls word vectors towards their initial (distributional) values, ensuring that relations encoded in initial vectors persist as long as they do not contradict the newly injected ones.

2.2 Language-Specific Rules and Constraints

Semantic Specialisation with Constraints

The fine-tuning ATTRACT-REPEL procedure is entirely driven by the input ATTRACT and REPEL sets of

| English | German | Italian |
|---------|--------|--------|
| (discuss, discussed) | (schottisch, schottischem) | (golfo, golfini) |
| (laugh, laughing) | (damalige, damalgen) | (metterere, mettetero) |
| (pacifist, pacifists) | (kombiniere, kombinierte) | (crescendo, cresci) |
| (evacuate, evacuated) | (schweigt, schweigt) | (crediti, crediti) |
| (evaluate, evaluates) | (hacken, gehackt) | (realtà, irrealtà) |
| (evacuate, evacuated) | (stabil, unstabil) | (realità, irrealità) |
| (laugh, laughing) | (formality, informality) | (attuato, inattuato) |

Table 2: Example synonymous (inflectional; top) and antonymous (derivational; bottom) constraints.
constraints. These can be extracted from a variety of semantic databases such as WordNet (Fellbaum, 1998), the Paraphrase Database (Ganitkevitch et al., 2013; Pavlick et al., 2015), or BabelNet (Navigli and Ponzetto, 2012; Ehrmann et al., 2014) as done in prior work (Faruqui et al., 2015; Wieting et al., 2015; Mrkšić et al., 2016, i.a.). In this work, we investigate another option: extracting constraints without curated knowledge bases in a spectrum of languages by exploiting inherent language-specific properties related to linguistic morphology. This relaxation ensures a wider portability of ATTRACT-REPEL to languages and domains without readily available or adequate resources.

**Extracting ATTRACT Pairs** The core difference between inflectional and derivational morphology can be summarised in a few lines as follows: the former refers to a set of processes through which the word form expresses meaningful syntactic information, e.g., verb tense, without any change to the semantics of the word. On the other hand, the latter refers to the formation of new words with semantic shifts in meaning (Schone and Jurafsky, 2001; Haspelmath and Sims, 2013; Lazaridou et al., 2013; Zeller et al., 2013; Cotterell and Schütze, 2017).

For the ATTRACT constraints, we focus on inflectional rather than on derivational morphology rules as the former preserve the full meaning of a word, modifying it only to reflect grammatical roles such as verb tense or case markers (e.g., (en_read, en_reads) or (de_katalanisch, de_katalanischer)). This choice is guided by our intent to fine-tune the original vector space in order to improve the embedded semantic relations.

We define two rules for English, widely recognised as morphologically simple (Avramidis and Koehn, 2008; Cotterell et al., 2016b).

| | W1 | A1 | B1 |
|---|---|---|---|
| English | 1,368,891 | 231,448 | 45,964 |
| German | 1,216,161 | 648,344 | 54,644 |
| Italian | 541,779 | 278,974 | 21,400 |
| Russian | 950,783 | 408,400 | 32,174 |

Table 3: Vocabulary sizes and counts of ATTRACT (A) and REPEL (R) constraints.

If $w_1$ ends with the letter e and $w_1 \in W_{en}$ and $w_2 \in W_{en}$, where $w_2 = w_1[-1:] + ing/ed$, then add $(w_1, w_2)$ and $(w_2, w_1)$ to the set of ATTRACT constraints $A$. This rule yields pairs such as (create, creating) and (create, created). Naturally, introducing more sophisticated rules is possible in order to cover for other special cases and morphological irregularities (e.g., sweep / swept), but in all our EN experiments, A is based on the two simple EN rules R1 and R2.

The other three languages, with more complicated morphology, yield a larger number of rules. In Italian, we rely on the sets of rules spanning: (1) regular formation of plural (libro / libri); (2) regular verb conjugation (aspettare / aspettiamo); (3) regular formation of past participle (aspettare / aspettato); and (4) rules regarding grammatical gender (bianco / bianca). Besides these, another set of rules is used for German and Russian: (5) regular declension (e.g., asiatisch / asiatischem).

**Extracting REPEL Pairs** As another source of implicit semantic signals, W also contains words which represent derivational antonyms: e.g., two words that denote concepts with opposite meanings, generated through a derivational process. We use a standard set of EN “antonymy” prefixes: $AP_{en} = \{\text{dis}, \text{il}, \text{un}, \text{in}, \text{im}, \text{ir}, \text{mis}, \text{non}, \text{anti}\}$ (Fromkin et al., 2013). If $w_1, w_2 \in W_{en}$, where $w_2$ is generated by adding a prefix from $AP_{en}$ to $w_1$, then $(w_1, w_2)$ and $(w_2, w_1)$ are added to the set of REPEL constraints $R$. This rule generates pairs such as (advantage, disadvantage) and (regular, irregular). An additional rule replaces the suffix -ful with -less, extracting antonyms such as (careful, careless).

Following the same principle, we use $AP_{de} = \{\text{un}, \text{nicht}, \text{anti}, \text{ir}, \text{in}, \text{miss}\}$, $AP_{it} = \{\text{in}, \text{ir}, \text{im}, \text{anti}\}$, and $AP_{ru} = \{\text{we}, \text{antti}\}$. For instance, this generates an IT pair (rispettoso, irrispettoso) (see Fig. 1). For DE, we use another rule targeting suffix replacement: -voll is replaced by -los.

We further expand the set of REPEL constraints by transitively combining antonymy pairs from the previous step with inflectional ATTRACT pairs. This step yields additional constraints such as (rispettoso, irrispettosi) (see Fig. 1). The final A and R constraint counts are given in Tab. 3. The full sets of rules are available as supplemental material.

3 Experimental Setup

**Training Data and Setup** For each of the four languages we train the skip-gram with negative sampling (SGNS) model (Mikolov et al., 2013)
on the latest Wikipedia dump of each language. We induce 300-dimensional word vectors, with the frequency cut-off set to 10. The vocabulary sizes $|W|$ for each language are provided in Tab. 3. We label these collections of vectors SGNS-LARGE.

**Other Starting Distributional Vectors** We also analyse the impact of morph-fitting on other collections of well-known EN word vectors. These vectors have varying vocabulary coverage and are trained with different architectures. We test standard distributional models: Common-Crawl GloVe (Pennington et al., 2014), SGNS vectors (Mikolov et al., 2013) with various contexts ($BOW = \text{bag-of-words}; DEPS = \text{dependency contexts}$), and training data ($PW = \text{Polyglot Wikipedia from Al-Rfou et al. (2013)}; 88 = 8 \text{ billion token word2vec corpus}$), following (Levy and Goldberg, 2014) and (Schwartz et al., 2015). We also test the symmetric-pattern based vectors of Schwartz et al. (2016) ($\text{SymPat-Emb}$), count-based PMI-weighted vectors reduced by SVD (Baroni et al., 2014) ($\text{Count-SVD}$), a model which replaces the context modelling function from CBOW with bidirectional LSTMs (Melamud et al., 2016) ($\text{Context2Vec}$), and two sets of EN vectors trained by injecting multilingual information: $\text{BiSkip}$ (Luong et al., 2015) and $\text{MultiCCA}$ (Faruqui and Dyer, 2014).

We also experiment with standard well-known distributional spaces in other languages (IT and DE), available from prior work (Dinu et al., 2015; Luong et al., 2015; Vulić and Korhonen, 2016a).

**Morph-fixed Vectors** A baseline which utilises an equal amount of knowledge as morph-fitting, termed morph-fixing, fixes the vector of each word to the distributional vector of its most frequent inflectional synonym, tying the vectors of low-frequency words to their more frequent inflections. For each word $w_1$, we construct a set of $M + 1$ words $W_{w_1} = \{w_1, w'_1, \ldots, w'_M\}$ consisting of the word $w_1$ itself and all $M$ words which co-occur with $w_1$ in the ATTRACT constraints. We then choose the word $w'_{\text{max}}$ from the set $W_{w_1}$ with the maximum frequency in the training data, and fix all other word vectors in $W_{w_1}$ to its word vector. The morph-fixed vectors (MFtX) serve as our primary baseline, as they outperformed another straightforward baseline based on stemming across all of our intrinsic and extrinsic experiments.

**Morph-fitting Variants** We analyse two variants of morph-fitting: (1) using ATTRACT constraints only (MFtX-A), and (2) using both ATTRACT and REPEL constraints (MFtX-AR).

### 4 Intrinsic Evaluation: Word Similarity

**Evaluation Setup and Datasets** The first set of experiments intrinsically evaluates morph-fitted vector spaces on word similarity benchmarks, using Spearman’s rank correlation as the evaluation metric. First, we use the SimLex-999 dataset, as well as SimVerb-3500, a recent EN verb pair similarity dataset providing similarity ratings for 3,500 verb pairs. SimLex-999 was translated to DE, IT, and RU by Leviant and Reichart (2015), and they crowdsourced similarity scores from native speakers. We use this dataset for our multilingual evaluation.

**Morph-fitting EN Word Vectors** As the first experiment, we morph-fit a wide spectrum of EN distributional vectors induced by various architectures (see Sect. 3). The results on SimLex and SimVerb are summarised in Tab. 4. The results with EN SGNS-LARGE vectors are shown in Fig. 3a. Morph-fitted vectors bring consistent improvement across all experiments, regardless of the quality of the initial distributional space. This finding confirms that the method is robust: its effectiveness does not depend on the architecture used to construct the initial space. To illustrate the improvements, note that the best score on SimVerb for a model trained on running text is achieved by $\text{Context2vec}$ ($\rho = 0.388$); injecting morphological constraints into this vector space results in a gain of 7.1 $\rho$ points.

**Experiments on Other Languages** We next extend our experiments to other languages, testing both morph-fitting variants. The results are summarised in Tab. 5, while Fig. 3a-3d show results for the morph-fitted SGNS-LARGE vectors. These scores confirm the effectiveness and robustness of morph-fitting across languages, suggesting that the idea of fitting to morphological constraints is indeed language-agnostic, given the set of languagespecific rule-based constraints. Fig. 3 also demon-

---

*Other SGNS parameters were set to standard values (Baroni et al., 2014; Vulić and Korhonen, 2016b): 15 epochs, 15 negative samples, global learning rate: .025, subsampling rate: $1e - 4$. Similar trends in results persist with $d = 100, 500$.*
We also tried using other post-processing spe-
This demonstrates that both types of constraints are
consistent gains over the initial spaces, and (con-
portant for the fine-tuning process. MF
strates that the morph-fitted vector spaces consis-
tently outperform the morph-fixed ones.

The comparison between MFIT-A and MFIT-
AR indicates that both sets of constraints are im-
portant for the fine-tuning process. MFIT-A yields
consistent gains over the initial spaces, and (con-
sistent) further improvements are achieved by also
incorporating the antonymous REPEL constraints.
This demonstrates that both types of constraints are
useful for semantic specialisation.

Comparison to Other Specialisation Methods
We also tried using other post-processing special-
isation models from the literature in lieu of
ATTRACTION-REPEL using the same set of “morpho-
logical” synonymy and antonymy constraints. We
compare ATTRACTION-REPEL to the retrofitting model
of (Faruqui et al., 2015) and counter-fitting (Mrkšič
et al., 2017a). The two baselines were trained for
20 iterations using suggested settings. The results
for EN, DE, and IT are summarised in Fig. 2. They
clearly indicate that MFIT-AR outperforms the two
other post-processors for each language. We hy-
pothesise that the difference in performance mainly
stems from context-sensitive vector space updates
performed by ATTRACTION-REPEL. Conversely, the
other two models perform pairwise updates which
do not consider what effect each update has on the
example pair’s relation to other word vectors (for a
detailed comparison, see (Mrkšič et al., 2017b)).

Besides their lower performance, the two other
specialisation models have additional disadvantages
compared to the proposed morph-fitting model.
First, retrofitting is able to incorporate
only synonymy/ATTRACTION pairs, while our
results demonstrate the usefulness of both types of
constraints, both for intrinsic evaluation (Tab. 5)
and downstream tasks (see later Fig. 3). Second,
counter-fitting is computationally intractable with
SGNS-LARGE vectors, as its regularisation term in-
volves the computation of all pairwise distances
between words in the vocabulary.

Further Discussion  The simplicity of the used
language-specific rules does come at a cost of occa-
sionally generating incorrect linguistic constraints
such as (tent, intent), (prove, improve) or (press, impress).
In future work, we will study how to fur-
ther refine extracted sets of constraints. We also plan to conduct experiments with gold standard morphological lexicons on languages for which such resources exist (Sylak-Glassman et al., 2015; Cotterell et al., 2016b), and investigate approaches which learn morphological inflections and derivations in different languages automatically as another potential source of morphological constraints (Soricut and Och, 2015; Cotterell et al., 2016a; Faruqui et al., 2016; Kann et al., 2017; Aharoni and Goldberg, 2017, i.a.).

5 Downstream Task: Dialogue State Tracking (DST)

Goal-oriented dialogue systems provide conversational interfaces for tasks such as booking flights or finding restaurants. In slot-based systems, application domains are specified using ontologies that define the search constraints which users can express. An ontology consists of a number of slots and their assorted slot values. In a restaurant search domain, sets of slot-values could include \texttt{PRICE} = \{cheap, expensive\} or \texttt{FOOD} = \{Thai, Indian, ...\}.

The DST model is the first component of modern dialogue pipelines (Young, 2010). It serves to capture the intents expressed by the user at each dialogue turn and update the belief state. This probability distribution over the possible dialogue states (defined by the domain ontology) is the system’s internal estimate of the user’s goals. It is used by the downstream dialogue manager component to choose the subsequent system response (Su et al., 2016). The following example shows the true dialogue state in a multi-turn dialogue:

\begin{itemize}
  \item \textbf{User}: What’s good in the southern part of town?
  \texttt{inform(area=south)}

  \item \textbf{System}: Vedanta is the top-rated Indian place.

  \item \textbf{User}: How about something cheaper?
  \texttt{inform(area=south, price=cheap)}

  \item \textbf{System}: Seven Days is very popular. Great hot pot.

  \item \textbf{User}: What’s the address?
  \texttt{inform(area=south, price=cheap); request(address)}

  \item \textbf{System}: Seven Days is at 66 Regent Street.
\end{itemize}

The Dialogue State Tracking Challenge (DSTC) shared task series formalised the evaluation and provided labelled DST datasets (Henderson et al., 2014a,b; Williams et al., 2016). While a plethora of DST models are available based on, e.g., hand-crafted rules (Wang et al., 2014) or conditional random fields (Lee and Eskenazi, 2013), the recent DST methodology has seen a shift towards neural-network architectures (Henderson et al., 2014c,d; Zilka and Jurcicek, 2015; Mrkšić et al., 2015; Perez and Liu, 2017; Liu and Perez, 2017; Vodolán et al., 2017; Mrkšić et al., 2017a, i.a.).

Model: Neural Belief Tracker To detect intents in user utterances, most existing models rely on either (or both): 1) Spoken Language Understanding models which require large amounts of annotated training data; or 2) hand-crafted, domain-specific lexicons which try to capture lexical and morphological variation. The Neural Belief Tracker (NBT) is a novel DST model which overcomes both issues by reasoning purely over pre-trained word vectors (Mrkšić et al., 2017a). The NBT learns to compose these vectors into intermediate utterance and context representations. These are then used to decide which of the ontology-defined intents (goals) have been expressed by the user. The NBT model keeps word vectors fixed during training, so that unseen, yet related words can be mapped to the right intent at test time (e.g. northern to north).

Data: Multilingual WOZ 2.0 Dataset Our DST evaluation is based on the WOZ dataset, released by Wen et al. (2017). In this Wizard-of-Oz setup, two Amazon Mechanical Turk workers assumed the role of the user and the system asking/providing information about restaurants in Cambridge (operating over the same ontology and database used for DSTC2 (Henderson et al., 2014a)). Users typed instead of speaking, removing the need to deal with noisy speech recognition. In DSTC datasets, users would quickly adapt to the system’s inability to deal with complex queries. Conversely, the WOZ setup allowed them to use sophisticated language. The WOZ 2.0 release expanded the dataset to 1,200 dialogues (Mrkšić et al., 2017a). In this work, we use translations of this dataset to Italian and German, released by Mrkšić et al. (2017b).

Evaluation Setup The principal metric we use to measure DST performance is the joint goal accuracy, which represents the proportion of test set dialogue turns where all user goals expressed up to that point of the dialogue were decoded correctly (Henderson et al., 2014a). The NBT models for EN, DE and IT are trained using four variants of the \texttt{SGNS-LARGE} vectors: 1) the initial distributional vectors; 2) \texttt{morph-fixed} vectors; 3) and 4) the two variants of \texttt{morph-fitted} vectors (see Sect. 3).

As shown by Mrkšić et al. (2017b), \textit{semantic specialisation} of the employed word vectors ben-
efits DST performance across all three languages. However, large gains on SimLex-999 do not always induce correspondingly large gains in downstream performance. In our experiments, we investigate the extent to which morph-fitting improves DST performance, and whether these gains exhibit stronger correlation with intrinsic performance.

**Results and Discussion**  The dark bars (against the right axes) in Fig. 3 show the DST performance of NBT models making use of the four vector collections. IT and DE benefit from both kinds of morph-fitting: IT performance increases from 74.1 → 78.1 (MFit-A) and DE performance rises even more: 60.6 → 66.3 (MFit-AR), setting a new state-of-the-art score for both datasets. The morph-fixed vectors do not enhance DST performance, probably because fixing word vectors to their highest frequency inflectional form eliminates useful semantic content encoded in the original vectors. On the other hand, morph-fitting makes use of this information, supplementing it with semantic relations between different morphological forms. These conclusions are in line with the SimLex gains, where morph-fitting outperforms both distributional and morph-fixed vectors.

English performance shows little variation across the four word vector collections investigated here. This corroborates our intuition that, as a morphologically simpler language, English stands to gain less from fine-tuning the morphological variation for downstream applications. This result again points at the discrepancy between intrinsic and extrinsic evaluation: the considerable gains in SimLex performance do not necessarily induce similar gains in downstream performance. Additional discrepancies between SimLex and downstream DST performance are detected for German and Italian. While we observe a slight drop in SimLex performance with the DE MFit-AR vectors compared to the MFit-A ones, their relative performance is reversed in the DST task. On the other hand, we see the opposite trend in Italian, where the MFit-A vectors score lower than the MFit-AR vectors on SimLex, but higher on the DST task. In summary, we believe these results show that SimLex is not a perfect proxy for downstream performance in language understanding tasks. Regardless, its performance does correlate with downstream performance to a large extent, providing a useful indicator for the usefulness of specific word vector
spaces for extrinsic tasks such as DST.

6 Related Work

Semantic Specialisation A standard approach to incorporating external information into vector spaces is to pull the representations of similar words closer together. Some models integrate such constraints into the training procedure, modifying the prior or the regularisation (Yu and Dredze, 2014; Xu et al., 2014; Bian et al., 2014; Kiela et al., 2015), or using a variant of the SGNS-style objective (Liu et al., 2015; Osborne et al., 2016). Another class of models, popularly termed retrofitting, injects lexical knowledge from available semantic databases (e.g., WordNet, PPDB) into pre-trained word vectors (Faruqui et al., 2015; Jauhar et al., 2015; Wieting et al., 2015; Nguyen et al., 2016; Mrkšić et al., 2016). Morph-fitting falls into the latter category. However, instead of resorting to curated knowledge bases, and experimenting solely with English, we show that the morphological richness of any language can be exploited as a source of inexpensive supervision for fine-tuning vector spaces, at the same time specialising them to better reflect true semantic similarity, and learning more accurate representations for low-frequency words.

Word Vectors and Morphology The use of morphological resources to improve the representations of morphemes and words is an active area of research. The majority of proposed architectures encode morphological information, provided either as gold standard morphological resources (Sylak-Glassman et al., 2015) such as CELEX (Baayen et al., 1995) or as an external analyser such as Morfessor (Creutz and Lagus, 2007), along with distributional information jointly at training time in the language modelling (LM) objective (Luong et al., 2013; Botha and Blunsom, 2014; Qiu et al., 2014; Cotterell and Schütze, 2015; Bhatia et al., 2016, i.a.). The key idea is to learn a morphological composition function (Lazaridou et al., 2013; Cotterell and Schütze, 2017) which synthesises the representation of a word given the representations of its constituent morphemes. Contrary to our work, these models typically coalesce all lexical relations.

Another class of models, operating at the character level, shares a similar methodology: such models compose token-level representations from sub-component embeddings (subwords, morphemes, or characters) (dos Santos and Zadrozny, 2014; Ling et al., 2015; Cao and Rei, 2016; Kim et al., 2016; Wieting et al., 2016; Verwimp et al., 2017, i.a.).

In contrast to prior work, our model decouples the use of morphological information, now provided in the form of inflectional and derivational rules transformed into constraints, from the actual training. This pipelined approach results in a simpler, more portable model. In spirit, our work is similar to Cotterell et al. (2016b), who formulate the idea of post-training specialisation in a generative Bayesian framework. Their work uses gold morphological lexicons; we show that competitive performance can be achieved using a non-exhaustive set of simple rules. Our framework facilitates the inclusion of antonyms at no extra cost and naturally extends to constraints from other sources (e.g., WordNet) in future work. Another practical difference is that we focus on similarity and evaluate morph-fitting in a well-defined downstream task where the artefacts of the distributional hypothesis are known to prompt statistical system failures.

7 Conclusion and Future Work

We have presented a novel morph-fitting method which injects morphological knowledge in the form of linguistic constraints into word vector spaces. The method makes use of implicit semantic signals encoded in inflectional and derivational rules which describe the morphological processes in a language. The results in intrinsic word similarity tasks show that morph-fitting improves vector spaces induced by distributional models across four languages. Finally, we have shown that the use of morph-fitted vectors boosts the performance of downstream language understanding models which rely on word representations as features, especially for morphologically rich languages such as German.

Future work will focus on other potential sources of morphological knowledge, porting the framework to other morphologically rich languages and downstream tasks, and on further refinements of the post-processing specialisation algorithm and the constraint selection.

Acknowledgments

This work is supported by the ERC Consolidator Grant LEXICAL: Lexical Acquisition Across Languages (no 648909). RR is supported by the Intel-ICRI grant: Hybrid Models for Minimally Supervised Information Extraction from Conversations. The authors are grateful to the anonymous reviewers for their helpful suggestions.
References

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In Proceedings of ACL. https://arxiv.org/abs/1611.01487.

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. In Proceedings of CoNLL, pages 183–192. http://www.aclweb.org/anthology/W13-3520.

Eleftherios Avramidis and Philipp Koehn. 2008. Enriching morphologically poor languages for statistical machine translation. In Proceedings of ACL, pages 763–770. http://www.aclweb.org/anthology/P/P08/P08-1087.

Harald R. Baayen, Richard Piepenbrock, and Hedderik van Rijn. 1995. The CELEX lexical data base on CD-ROM.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In Proceedings of ACL, pages 809–815. http://www.aclweb.org/anthology/P14-2131.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In Proceedings of ACL, pages 238–247. http://www.aclweb.org/anthology/P14-1023.

Parminder Bhatia, Robert Guthrie, and Jacob Eisenstein. 2016. Morphological priors for probabilistic neural word embeddings. In Proceedings of EMNLP, pages 490–500. https://aclweb.org/anthology/D16-1047.

Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Knowledge-powered deep learning for word embedding. In Proceedings of ECML-PKDD, pages 132–148. https://doi.org/10.1007/978-3-662-44848-9_9.

Jan A. Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. In Proceedings of ICML, pages 1899–1907. http://jmlr.org/proceedings/papers/v32/botha14.html.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. Journal of Artificial Intelligence Research 49:1–47. https://doi.org/10.1613/jair.4135.

Kris Cao and Marek Rei. 2016. A joint model for word embedding and word morphology. In Proceedings of the 1st Workshop on Representation Learning for NLP, pages 18–26. http://aclweb.org/anthology/W16/W16-1603.

Danqi Chen and Christopher D. Manning. 2014. A fast and accurate dependency parser using neural networks. In Proceedings of EMNLP, pages 740–750. http://www.aclweb.org/anthology/D14-1082.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural language processing (almost) from scratch. Journal of Machine Learning Research 12:2493–2537. http://dl.acm.org/citation.cfm?id=1953048.2078186.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016a. The sigmorphon 2016 shared task - morphological reinflection. In Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 10–22. http://anthology.aclweb.org/W16-2002.

Ryan Cotterell and Hinrich Schütze. 2015. Morphological word-embeddings. In Proceedings of NAACL-HLT, pages 1287–1292. http://www.aclweb.org/anthology/N15-1140.

Ryan Cotterell and Hinrich Schütze. 2017. Joint semantic synthesis and morphological analysis of the derived word. Transactions of the ACL. https://arxiv.org/abs/1701.00946.

Ryan Cotterell, Hinrich Schütze, and Jason Eisner. 2016b. Morphological smoothing and extrapolation of word embeddings. In Proceedings of ACL, pages 1651–1660. http://www.aclweb.org/anthology/P16-1156.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. TSLP 4(1):3:1–3:34. http://doi.acm.org/10.1145/1217098.1217101.

James Curran. 2004. From Distributional to Semantic Similarity. Ph.D. thesis, School of Informatics, University of Edinburgh. http://hdl.handle.net/1842/563.

Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. 2015. Improving zero-shot learning by mitigating the hubness problem. In Proceedings of ICLR (Workshop Papers). http://arxiv.org/abs/1412.6568.

Cicero Nogueira dos Santos and Bianca Zadrozny. 2014. Learning character-level representations for part-of-speech tagging. In Proceedings of ICML, pages 1818–1826. http://jmlr.org/proceedings/papers/v32/santos14.html.

John C. Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. Journal of Machine Learning Research 12:2121–2159. http://dl.acm.org/citation.cfm?id=2021068.

Maud Ehrmann, Francesco Coccioni, Daniele Vannella, John Philip Mccrae, Philipp Cimiano, and Roberto Navigli. 2014. Representing multilingual data as linked data: The case of BabelNet 2.0. In Proceedings of LREC, pages 401–408. http://www.lrec-conf.org/proceedings/lrec2014/summaries/810.html.
Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In Proceedings of NAACL-HLT. pages 1606–1615. http://www.aclweb.org/anthology/N15-1184.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In Proceedings of EACL. pages 462–471. http://www.aclweb.org/anthology/E14-1049.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In Proceedings of NAACL-HLT. pages 634–643. http://www.aclweb.org/anthology/N16-1077.

Christiane Fellbaum. 1998. WordNet. https://mitpress.mit.edu/books/wordnet.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. ACM Transactions on Information Systems 20(1):116–131. https://doi.org/10.1145/503104.503110.

Victoria Fromkin, Robert Rodman, and Nina Hyams. 2013. An Introduction to Language, 10th Edition.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. In Proceedings of NAACL-HLT. pages 758–764. http://www.aclweb.org/anthology/N13-1092.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A large-scale evaluation set of verb similarity. In Proceedings of EMNLP. pages 2173–2182. https://aclweb.org/anthology/D16-1235.

Zellig S. Harris. 1954. Distributional structure. Word 10(23):146–162.

Martin Hasepflacht and Andrea Sims. 2013. Understanding morphology.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014a. The Second Dialog State Tracking Challenge. In Proceedings of SIGDIAL. pages 263–272. http://aclweb.org/anthology/W14/W14-4337.pdf.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014b. The Third Dialog State Tracking Challenge. In Proceedings of IEEE SLT. pages 324–329. https://doi.org/10.1109/SLT.2014.7078595.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014c. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation. In Proceedings of IEEE SLT. pages 360–365.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014d. Word-based dialog state tracking with recurrent neural networks. In Proceedings of SIGDIAL. pages 292–299. http://aclweb.org/anthology/W/W14/W14-4340.pdf.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics 41(4):665–695. https://doi.org/10.1116/CO/Li_a_00237.

Sujay Kumar Jauhar, Chris Dyer, and Eduard H. Hovy. 2015. Ontologically grounded multi-sense representation learning for semantic vector space models. In Proceedings of NAACL. pages 683–693. http://www.aclweb.org/anthology/N15-1070.

Anders Johannsen, Héctor Martínez Alonso, and Anders Søgaard. 2015. Any-language frame-semantic parsing. In Proceedings of EMNLP. pages 2062–2066. http://aclweb.org/anthology/D15-1245.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017. Neural multi-source morphological reinflection. In Proceedings of EACL. pages 514–524. http://www.aclweb.org/anthology/E17-1049.

Douwe Kiela, Felix Hill, and Stephen Clark. 2015. Specializing word embeddings for similarity or relatedness. In Proceedings of EMNLP. pages 2044–2048. http://aclweb.org/anthology/D15-1242.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. 2016. Character-aware neural language models. In Proceedings of AAAI. pages 2741–2749.

Angeliki Lazaridou, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2013. Compositionally derived representations of morphologically complex words in distributional semantics. In Proceedings of ACL. pages 1517–1526. http://www.aclweb.org/anthology/P13-1149.

Sungjin Lee and Maxine Eskenazi. 2013. Recipe for building robust spoken dialog state trackers: Dialog State Tracking Challenge system description. In Proceedings of SIGDIAL. pages 414–422. http://aclweb.org/anthology/W/W13/W13-4066.pdf.

Ira Leviant and Roi Reichart. 2015. Separated by an un-common language: Towards judgment language informed vector space modeling. CoRR abs/1508.00106. http://arxiv.org/abs/1508.00106.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In Proceedings of ACL. pages 302–308. http://aclweb.org/anthology/P14-2050.

Wang Ling, Chris Dyer, Alan W. Black, Isabel Trancoso, Ramon Fernandez, Silvio Amir, Luis Marujo, and Tiago Luis. 2015. Finding function in form:
Compositional character models for open vocabulary word representation. In Proceedings of EMNLP, pages 1520–1530. http://aclweb.org/anthology/D15-1176.

Fei Liu and Julien Perez. 2017. Gated end-to-end memory networks. In Proceedings of EACL, pages 1–10. http://www.aclweb.org/anthology/E17-1001.

Quan Liu, Hui Jiang, Si Wei, Zhen-Hua Ling, and Yu Hu. 2015. Learning semantic word embeddings based on ordinal knowledge constraints. In Proceedings of ACL, pages 1501–1511. http://www.aclweb.org/anthology/P15-1145.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, pages 151–159. http://www.aclweb.org/anthology/W15-1521.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better word representations with recursive neural networks for morphology. In Proceedings of CoNLL. pages 104–113. http://www.aclweb.org/anthology/W13-3512.

Oren Melamud, Jacob Goldberger, and Ido Da- gan. 2016. Context2vec: Learning generic context embedding with bidirectional LSTM. In Proceedings of CoNLL, pages 51–61. http://aclweb.org/anthology/K/K16/K16-1006.pdf.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Proceedings of NIPS. pages 3111–3119.

Andriy Mnih and Koray Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In Proceedings of NIPS. pages 2265–2273.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multidomain dialog state tracking using recurrent neural networks. In Proceedings of ACL, pages 794–799. http://aclweb.org/anthology/P/P15/P15-2130.pdf.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Tsung-Hsien Wen, and Steve Young. 2017a. Neural Belief Tracker: Data-driven dialogue state tracking. In Proceedings of ACL. http://arxiv.org/abs/1606.03777.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina Maria Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In Proceedings of NAACL-HLT. http://aclweb.org/anthology/N/N16/N16-1018.pdf.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017b. Semantic Specialisation of Distributional Word Vector Spaces using Monolingual and Cross-Lingual Constraints. arXiv.

Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted Boltzmann machines. In Proceedings of ICML. pages 807–814. http://www.icml2010.org/papers/432.pdf.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial Intelligence 193:217–250. https://doi.org/10.1016/j.artint.2012.07.001.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2016. Integrating distributional lexical contrast into word embeddings for antonym-synonym distinction. In Proceedings of ACL, pages 454–459. http://aclweb.org/anthology/P16-2074.

Dominique Osborne, Shashi Narayan, and Shay Cohen. 2016. Encoding prior knowledge with eigenword embeddings. Transactions of the ACL 4:417–430. https://arxiv.org/abs/1509.01007.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In Proceedings of ACL, pages 425–430. http://www.aclweb.org/anthology/P15-2070.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In Proceedings of EMNLP. pages 1532–1543. http://www.aclweb.org/anthology/D14-1162.

Julien Perez and Fei Liu. 2017. Dialog state tracking, a machine reading approach using Memory Network. In Proceedings of EACL. pages 305–314. http://www.aclweb.org/anthology/E17-1029.

Siyu Qiu, Qing Cui, Jiang Bian, Bin Gao, and Tie-Yan Liu. 2014. Co-learning of word representations and morpheme representations. In Proceedings of COLING, pages 141–150. http://www.aclweb.org/anthology/C14-1015.

Patrick Schone and Daniel Jurafsky. 2001. Knowledge-free induction of inflectional morphologies. In Proceedings of NAACL. http://aclweb.org/anthology/N/N01/N01-1024.

Roy Schwartz, Roi Reichart, and Ari Rappoport. 2015. Symmetric pattern based word embeddings for improved word similarity prediction. In Proceedings of CoNLL. pages 258–267. http://www.aclweb.org/anthology/K15-1026.

Roy Schwartz, Roi Reichart, and Ari Rappoport. 2016. Symmetric patterns and coordinations: Fast and enhanced representations of verbs and adjectives.
Morph-fitting: Fine-Tuning Word Vector Spaces with Simple Language-Specific Rules
Supplementary Material

Morphological Rules

In this supplemental material, we provide a short comprehensive overview of simple language-specific morphological rules in English (EN), German (DE), Italian (IT), and Russian (RU). These rules were used to build the sets of synonymous ATTRACT and antonymous REPEL constraints for our morph-fitting fine-tuning procedure. As discussed in the paper, the linguistic constraints extracted from the rules require only a comprehensive list of vocabulary words in each language. A native speaker of each language used in our experiments is able to easily come up with these sets of morphological rules (or at least with a reasonable subset of rules) without any linguistic training. What is more, the rules for German, Italian, and Russian were created by non-native and non-fluent speakers who have only a passive or limited knowledge of the three languages, exemplifying the simplicity and portability of the fine-tuning approach based on the shallow “morphological supervision". The simplicity is also confirmed by the short time used to compile the rules, ranging from a few minutes for English to approximately two hours for Russian.

Different languages differ in their “morphological richness" (e.g., declension, verb conjugation, plural forming, gender) which consequently leads to the varying number of rules in each language. However, all four languages in our study display morphological regularities described by simple morphological rules that are exploited to build sets of ATTRACT and REPEL linguistic constraints in each language from scratch.\(^9\)

Vocabularies \(W\) in all four languages are labeled \(W_{en}\), \(W_{de}\), \(W_{it}\), \(W_{ru}\). We add the pairs \((w_1, w_2)\) and \((w_2, w_1)\) generated by the rules to the sets of constraints iff both \(w_1, w_2 \in W\). After we generate all such constraints, since some constraints may have been generated by more than one rule, we remove all duplicates from the respective sets of ATTRACT and REPEL constraints.

Before we start, we will define two simple functions: (i) the function \(w[: -N]\) strips the last \(N\) characters from the word \(w\), (ii) the function \(w.ew(\text{sub})\) tests if the word \(w\) ends with a sequence of characters \(\text{sub}\). For instance, \(\text{create}[: -1]\) returns \(\text{creat}\), while \(\text{create.ew}('s')\) returns \(\text{False}\) and \(\text{create.ew}('e')\) returns \(\text{True}\).

English Rules

Inflectional Synonymy: ATTRACT As discussed in the paper, we rely on only two simple inflectional morphological rules in English:
- \(w_2 = w_1 + 's'/ 'ed'/ 'ing'\). This rule yields constraints such as (\(\text{speaking}, \text{speaking}\), (\(\text{turtle}, \text{turtles}\), or (\(\text{clean}, \text{cleaned}\).
- If \(w_1.ew('e')\), then \(w_2 = w_1[: -1] + 'ed'/ 'ing'\). This rule yields constraints such as (\(\text{create}, \text{creating}\), or (\(\text{generate}, \text{generated}\).

Derivational Antonymy: REPEL We assume the following set of standard “antonymy" prefixes in English: \(\text{AP}_{en} = \{\text{dis' }, \text{il', un', in', im', ir', mis', non', anti'}\}.\) We rely on the following derivational rules to extract REPEL pairs:
- \(w_2 = \text{ap} + w_1\), where \(\text{ap} \in \text{AP}_{en}\). This rule yields constraints such as (\(\text{mature}, \text{immature}\), (\(\text{allow}, \text{disallow}\)) or (\(\text{regularity}, \text{irregularity}\).
- If \(w_1.ew('ful')\), then \(w_2 = w_1[: -3] + 'less'\). This rule yields constraints such as (\(\text{cheerful}, \text{cheerless}\).

As mentioned in the paper, for all four languages we further expand the set of REPEL constraints by transityely combining antonymy pairs with inflectional ATTRACT pairs. In simple words, the friend of my enemy is my enemy. This means that, given an ATTRACT pair (\(\text{allow}, \text{allows}\)) and a REPEL pair (\(\text{allow}, \text{disallow}\)), we extract another REPEL pair (\(\text{allows}, \text{disallow}\).

German Rules

Inflectional Synonymy: ATTRACT Being morphologically richer than English, the German language naturally requires more rules to describe its (inflectional) morphological richness and variation. First, we capture the regular declension of nouns and adjectives by the following heuristic:
- Generate a set of words \(W_{w_1} = \{w_1, w_2|w_2 = w_1 + 'e'/ 'em'/ 'en'/ 'er'/ 'es'\}\); take the Cartesian product on \(W_{w_1} \times W_{w_1}\) and then exclude \((w_1, w_1)\)

\(^9\)Note that the rules for extracting ATTRACT constraints were additionally used to generate the Morph-SimLex evaluation set, also provided as supplemental material.
pairs with identical words. This rule generates pairs such as (schottisch, schottische), (schottischem, schottischen).

The second set of rules describes regular verb morphology, i.e., verb conjugation in the present and past tense, and the formation of regular past participles. This set of rules may be expressed as:
- If \( w_1 \cdot \text{ew}('en') \), then \( w'_1 = w_1[1: -2] \).
- If \( w'_1 \cdot \text{ew}('t') \), then generate a set of words \( W_{w_1} = \{w_1, w_2[w_2 = w_1 + 'e' \text{'st'} \text{'te'} \text{'tet'} \text{'eten'}, w_2 = 'ge' + w'_1 + 'et' \} \), else (if not \( w'_1 \cdot \text{ew}('t') \)), generate a set of words \( W_{w_1} = \{w_1, w_2[w_2 = w_1 + 'e' \text{'st'} \text{'te'} \text{'tet'} \text{'eten'}, w_2 = 'ge' + w'_1 + 'et' \} \).

We then take the Cartesian product on \( W_{w_1} \times W_{w_1} \). Again, all pairs with identical words were discarded. This rule yields pairs such as (machen, machten), (mache, gemacht), (kaufst, kauft), or (arbeite, arbeite) and (arbeiten, gearbeitet).

Another set of rules targets the regular formation of plural nouns:
- If \( w_1 \cdot \text{ew}('ei') \) or \( w_1 \cdot \text{ew}('heit') \) or \( w_1 \cdot \text{ew}('keit') \) or \( w_1 \cdot \text{ew}('schaft') \) or \( w_1 \cdot \text{ew}('ung') \), then \( w_2 = w_1 + 'en' \). This rule yields pairs such as (wahrheit, wahrheiten) or (gemeinschaft, gemeinschaften).
- If \( w_1 \cdot \text{ew}('in') \), then \( w_2 = w_1 + 'nen' \). This rule generates pairs such as (lehrerin, lehrerinnen) or (lektorin, lektorinnen).
- If \( w_1 \cdot \text{ew}('a'/'e'/'o'/'i') \) then \( w_2 = w_1 + 's'. \) This rule yields rules such as (auto, autos).
- If \( w_1 \cdot \text{ew}('e') \), then \( w_2 = w_1 + 'n'. \) This rule yields pairs such as (postkarte, postkarten).

Derivational Antonymy: REPEL. We assume the following set of standard “antonymy” prefixes in German: \( AP_{\text{de}} = \{'un', 'nicht', 'anti', 'ir', 'in', 'miss'\}. \) We rely on the following derivational rules to extract REPEL pairs in German:
- \( w_2 = \text{ap} + w_1 \), where \( \text{ap} \in AP_{\text{de}}. \) This rule yields constraints such as (aktiv, inaktiv), (wandelbar, unwandelbar) or (zyklone, antizykline).
- If \( w_1 \cdot \text{ew}('voll') \), then \( w_2 = w_1[1: -4] + 'los'. \) This rule yields constraints such as (geschmackvoll, geschmacklos).

The set of REPEL is then again transitively expanded yielding pairs such as (relevant, irrelevant) or (aktivem, inaktiv).

Italian Rules

Inflectional Synonymy: ATTRACT. The first set of rules aims at capturing the regular plural forming in Italian (e.g., libro, libri) and regular differences in gender (e.g., rapido, rapida). We rely on the simple heuristic which can be expressed as follows:
- If \( w_1 \cdot \text{ew}('a'/'e'/'o'/'i') \), then generate a set of words \( W_{w_1} = \{w_2[w_2 = w_1[1:] -1] + 'a'/'e'/'o'/'i' \} \), and take the Cartesian product on \( W_{w_1} \times W_{w_1} \). This rule yields pairs such as (nero, neri) or (generazione, generazioni).
- If \( w_1 \cdot \text{ew}('go') \), then \( w_2 = w_1 + 'h'. \) This rule generates pairs such as (tartaruga, tartarughe) or (bianca, bianche).
- If \( w_1 \cdot \text{ew}('oro') \), then \( w_2 = w_1 + 'hi'. \) This rule generates pairs such as (albergo, alberghi).

The second set of rules targets regular verb conjugation in Italian and the formation of regular past participles. The following rules are used:
- If \( w_1 \cdot \text{ew}('are') \), then generate a set of words \( W_{w_1} = \{w_1, w_2[w_2 = w_1[1:] -3] + 'iamo'/'ete'/'ono'/'i' \} \) and regular differences in gender (e.g., rapido, rapida).
- If \( w_1 \cdot \text{ew}('ere') \), then generate a set of words \( W_{w_1} = \{w_1, w_2[w_2 = w_1[1:] -3] + 'iamo'/etu'/'ono'/io'/i' \} \) and regular differences in gender (e.g., rapido, rapida).
- If \( w_1 \cdot \text{ew}('ire') \), then generate a set of words \( W_{w_1} = \{w_1, w_2[w_2 = w_1[1:] -3] + 'iamo'/ite'/'ono'/io'/i' \} \) and regular differences in gender (e.g., rapido, rapida).

Derivational Antonymy: REPEL. We assume the following set of standard “antonymy” prefixes in Italian: \( AP_{\text{it}} = \{'in', 'ir', 'in', 'anti'\}. \) The following derivational rule is used to extract REPEL pairs:
- \( w_2 = \text{ap} + w_1 \), where \( \text{ap} \in AP_{\text{it}}. \) This rule yields constraints such as (attivo, inattivo) or (rispettosa, irrispettosa).
The set of REPEL was then expanded as before, e.g., with additional pairs such as (rispettosa, irrispettosi) generated.

**Russian Rules**

**Inflectional Synonymy: ATTRACTION**  The first set of rules in Russian targets the regular forming of plural in Russian. A few simple heuristics are used as follows:

- \( w_2 = w_1 + 'и' / 'ы' \). This rule yields pairs such as (альбом, альбомы), transliterated as: (al’bom, al’bomyi).
- if \( w_1 \in \{ 'а' / 'я' / 'ь' \} \), then \( w_2 = w_1: -1 \) + 'и' / 'ы'. This rule generates pairs such as (писня, песни): (pes’nia, pesni).
- if \( w_1 \in \{ 'о' \} \), then \( w_2 = w_1: -1 \) + 'а'. This rule generates pairs such as (пишьмо, письма): (pis’mo, pis’ma).
- if \( w_1 \in \{ 'е' \} \), then \( w_2 = w_1: -1 \) + 'я'. This rule generates pairs such as (платье, пла́тья): (plat’e, plat’ya).

The next set of rules targets regular verb conjugation of Russian verbs as well as the regular formation of past participles. We again build a simple heuristic to extract \( \text{ATTRACTION} \) pairs:

- if \( w_1 \in \{ 'тн' / 'тн' \} \), then generate a set of words \( W_{w_1} = \{ w_1, w_2 \} \), then generate a set of words \( W_{w_2} = \{ w_2 \} \) and take the Cartesian product on \( W_{w_1} \times W_{w_2} \) discarding pairs with identical words. This rule yields pairs such as (работам, работами): (rabotam, rabotami).
- if \( w_1 \in \{ 'ый' / 'ый' / 'й' \} \), then generate a set of words \( W_{w_1} = \{ w_1, w_2 | w_2 = w_1: -1 \} + 'й' \) and take the Cartesian product on \( W_{w_1} \times W_{w_2} \) discarding pairs with identical words. This rule yields pairs such as (работам, работами): (rabotam, rabotami).

Yet another set of rules targets regular adjective comparison and gender:

- if \( w_1 \in \{ 'ый' / 'ые' / 'ая' \} \), then generate a set of words \( W_{w_1} = \{ w_1, w_2 | w_2 = w_1: -2 \} + 'е' \) and take the Cartesian product on \( W_{w_1} \times W_{w_2} \) discarding pairs with identical words. This rule yields pairs such as (быстрый, быстрее): (bystroyj, bystroye).
- if \( w_1 \in \{ 'ой' \} \), then generate a set of words \( W_{w_1} = \{ w_1, w_2 | w_2 = w_1: -2 \} + 'е' \) and take the Cartesian product on \( W_{w_1} \times W_{w_2} \) discarding pairs with identical words. This rule yields pairs such as (новьё, новье): (novyjo, novyje).

**Derivational Antonymy: REPEL**  We assume the following set of standard “antonymy” prefixes in Russian: \( \text{AP}_1 = \{ 'не', анти' \} \), and simply use the following rule:

- \( w_2 = a \text{p} + w_1 \), where \( a \text{p} \in \text{AP}_1 \). This rule yields constraints such as (адекватный, неадекватный) or (вирусная, антивирусная), transiterated as: (adekvatnyj, neadekvatnyj) and (virusnaja, antivirusnaja).

The further expansion of REPEL constraints yields pairs such as (адекватный, неадекватный): (adekvatnyj, neadekvatnyj).

**Further Discussion**

We stress that the listed rules for all four languages are non-exhaustive and do not cover all possible inflectional and derivational morphological phenomena. More linguistic constraints may be extracted by resorting to more sophisticated rules covering finer-grained morphological processes (e.g., covering irregular plural forming or irregular verb conjugation and past participle forming, or non-standard declensions). Further, the listed rules, written by non-native speakers without any linguistic training in a very short time span, do not necessarily rely on established linguistic theories in each language, but are rather simple heuristics aiming to capture morphological regularities.