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

1 Morphological Rules

In this supplemental material, we provide a short comprehensive overview of simple language-specific morphological rules in English (EN), German (DE), Italian (UT), 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.[1]

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(sub)` tests if the word w ends with a sequence of characters `sub`. For instance, $create[:\ -1]$ returns $creat$, while $create$`.ew('s')` returns `False` and $create$`.ew('e')` returns `True`.

1.1 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 *(speak, speaking)*, *(turtle, turtles)*, or *(clean, cleaned)*.
- If w_1`.ew('e')`, then $w_2 = w_1[:\ -1]$ + `'ed'/'ing'`. This rule yields constraints such as *(create, creating)*, or *(generate, generated)*.

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

As mentioned in the paper, for all four languages we further expand the set of REPEL constraints by transitively 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 *(allow, allows)* and a REPEL pair *(allow, disallow)*, we extract another REPEL pair *(allows, disallow)*.

1.2 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_i, w_i) pairs with identical words. This rule generates

---

[1] Note that the rules for extracting ATTRACT constraints were additionally used to generate the Morph-SimLex evaluation set, also provided as supplemental material.

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 $w_1\texttt{.ew('en')}$, then $w_1' = w_1[: -2]$. If $w_1'\texttt{.ew('t')}$, then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1 + \text{'e'/'st'/'ete'/'etest'/'etet'/'eten'}, w_2 = \text{'ge'} + w_1' + \text{'et'}\}$, else (if not $w_1'\texttt{.ew('t')}$), generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1 + \text{'e'/'st'/'te'/'test'/'tet'/'ten'}, w_2 = \text{'ge'} + w_1' + \text{'t'}\}$. 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, arbeitete)* and *(arbeiten, gearbeitet)*.

Another set of rules targets the regular formation of *plural* nouns:

- If $w_1\texttt{.ew('ei')}$ or $w_1\texttt{.ew('heit')}$ or $w_1\texttt{.ew('keit')}$ or $w_1\texttt{.ew('schaft')}$ or $w_1\texttt{.ew('ung')}$, then $w_2 = w_1 + \texttt{'en'}$. This rule yields pairs such as *(wahrheit, wahrheiten)* or *(gemeinschaft, gemeinschaften)*.
- If $w_1\texttt{.ew('in')}$, then $w_2 = w_1 + \texttt{'nen'}$. This rule generates pairs such as *(lehrerin, lehrerinnen)* or *(lektorin, lektorinnen)*.
- If $w_1\texttt{.ew('a'/'i'/'o'/'u'/'y')}$ then $w_2 = w_1 + \texttt{'s'}$. This rule yields pairs such as *(auto, autos)*.
- If $w_1\texttt{.ew('e')}$, then $w_2 = w_1 + \texttt{'n'}$. This rule yields pairs such as *(postkarte, postkarten)*.
- $w_2 = \mathit{lumlaut}(w_1) + \texttt{er}$, where the function $\mathit{lumlaut}(w)$ replaces the last occurrence of the letter `'a'`,`'o'` or `'u'` with `'ä'`,`'ö'` or `'ü'`. This rule generates pairs such as *(wörterbuch, wörterbücher)* or *(stadt, städter)*.

Derivational Antonymy: REPEL We assume the following set of standard "antonymy" prefixes in German: AP_{de} = *{'un', 'nicht', 'anti', 'ir', 'in', 'miss'}*. We rely on the following derivational rules to extract REPEL pairs in German:

- $w_2 = \texttt{ap} + w_1$, where $\texttt{ap} \in AP_{de}$. This rule yields constraints such as *(aktiv, inaktiv)*, *(wandelbar, unwandelbar)* or *(zyklone, antizyklone)*.
- If $w_1\texttt{.ew('voll')}$, then $w_2 = w_1[: -4] + \texttt{'los'}$. This rule yields constraints such as *(geschmackvoll, geschmacklos)*.

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

1.3 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 $w_1\texttt{.ew('a'/'e'/'o'/'i')}$, then generate a set of words $W_{w_1} = \{w_2 | w_2 = w_1[: -1] + \text{'a'/'e'/'o'/'i'}\}$, and take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule yields pairs such as *(nero, neri)* or *(generazione, generazioni)*.
- If $w_1\texttt{.ew('ga'/'ca')}$, then $w_2 = w_1 + \texttt{'he'}$. This rule generates pairs such as *(tartaruga, tartarughe)* or *(bianca, bianche)*.
- If $w_1\texttt{.ew('go')}$, then $w_2 = w_1 + \texttt{'hi'}$. 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 $w_1\texttt{.ew('are')}$, then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -3] + \text{'iamo'/'ate'/'ano'/'o'/'i'/'a'/'ato'/'ata'/'ati'/'ate'}\}$; take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule results in pairs such as *(aspettare, aspettiamo)*.
- If $w_1\texttt{.ew('ere')}$, then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -3] + \text{'iamo'/'ete'/'ono'/'o'/'i'/'e'/'uto'/'uta'/'uti'/'ute'}\}$; take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule results in pairs such as *(ricevere, ricevete)* or *(riceve, ricevuto)*.
- If $w_1\texttt{.ew('ire')}$, then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -3] + \text{'iamo'/'ite'/'ono'/'o'/'i'/'e'/'ito'/'ita'/'iti'/'ite'}\}$; take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule results in pairs such as *(dormire, dormono)* or *(dormi, dormita)*.

Derivational Antonymy: REPEL We assume the following set of standard "antonymy" prefixes in Italian: AP_{it} = *{'in', 'ir', 'im', 'anti'}*. We rely on the following derivational rules to extract REPEL pairs in Italian:

- $w_2 = \texttt{ap} + w_1$, where $\texttt{ap} \in AP_{de}$. 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.

1.4 Russian Rules

Inflectional Synonymy: ATTRACT 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'bomy)*.

- if w_1.ew('а'/'я'/'ь'), then $w_2 = w_1[: -1]$ + 'и'/'ы'. This rule generates pairs such as (песня, песни): *(pesnja, pesni)*.

- if w_1.ew('о'), then $w_2 = w_1[: -1]$ + 'а'. This rule generates pairs such as (письмо, письма): *(pis'mo, pis'ma)*.

- if w_1.ew('е'), then $w_2 = w_1[: -1]$ + 'я'. This rule generates pairs such as (платье, платья): *(plat'e, plat'ja)*.

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 ATTRACT pairs:

- if w_1.ew('ти'/'ть'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -2] + \text{'у'/'ю'/'ешь'/'ишь'/'ет'/'ит'/'ем'/'им'}, w_2 = w_1[: -2] + \text{'ете'/ите'/'ут'/'ют'/'ат'/'ят'}, w_2 = w_1[: -2] + \text{'нный'/'нная'}\}$ and take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule yields pairs such as (варить, варите) or (заканчиваю, заканчивают), transliterated as: *(varit', varite)*, *(zakanchivaju, zakanchivajut)*.

Following that, we also utilise the regularities regarding declension processes in Russian, captured by the following rules:

- if w_1.ew('а'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -1] + \text{'е'/'у'/'ой'}\}$ and take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule yields pairs such as (работа, работой): *(rabota, rabotoj)*.

- if w_1.ew('я'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -1] + \text{'е'/'ю'/'ей'}\}$ and take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule yields pairs such as (линия, линию): *(linija, liniju)*.

- if w_1.ew('ы'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -1] + \text{'ам'/'ами'/'ах'}\}$ and take the Cartesian product on $W_{w_1} \times W_{w_1}$ discarding pairs with identical words. This rule yields pairs such as (работам, работами): *(rabotam, rabotami)*.

- if w_1.ew('и'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -1] + \text{'ь'/'ям'/'ями'/'ях'}\}$ and take the Cartesian product on $W_{w_1} \times W_{w_1}$ 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.ew('ый'/'ой'/'ий'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -2] + \text{'ь'/'ее'/'ые'}\}$. This rule yields pairs such as (быстрый, быстрее): *(bystryj, bystree)*.

- if w_1.ew('ая'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -2] + \text{'ее'/'ые'/'ый'}\}$. This rule yields pairs such as (новая, новые): *(novaja, novye)*.

- if w_1.ew('ое'), then generate a set of words $W_{w_1} = \{w_1, w_2 | w_2 = w_1[: -2] + \text{'ый'/'ые'/'ая'}\}$. This rule yields pairs such as (новое, новый): *(novoe, novyj)*.

Derivational Antonymy: REPEL We assume the following set of standard "antonymy" prefixes in Russian: AP_{ru} = *{*не, анти*'}*, and simply use the following rule:

- w_2 = ap + w_1, where ap $\in AP_{ru}$. This rule yields constraints such as (адекватный, неадекватный) or (вирусная, антивирусная), transliterated as: *(adekvatnyj, neadekvatnyj)* and *(virusnaja, antivirusnaja)*.

The further expansion of REPEL constraints yields pairs such as (адекватный, неадекватная): *(adekvatnyj, neadekvatnaja)*.

1.5 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.