AUTOMATIC GENERATION OF TUNABLE ANALOGY BENCHMARKS FOR WORD REPRESENTATIONS

BY

TAREK J. SAKAKINI

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Advisers:

Professor Pramod Viswanath
Research Assistant Professor Suma Bhat

ABSTRACT

We present a method to automatically generate syntactic analogy datasets for the evaluation of word representations in an unsupervised manner. The automatic generation also allows for customization in terms of word-frequencies, syntactic rules, part-of-speech tags and size of the dataset. We show the ability of our method to generate cross-lingual analogy task datasets for languages other than English, where evaluation datasets are limited if not nonexistent, by constructing datasets for French, German, Spanish, Arabic and Hebrew.

Our method clusters pairs of words into morphological rules in an unsupervised manner, using which we generate analogy questions for different rules. We show the quality of an automatically generated dataset by checking the correlation of the performance of different word representations on it with the performance of the same representations on the Google analogy dataset. The values exhibited a high correlation of 95%. Moreover, we showcase the benefits of customization through studying the performance of different word representations when varying the frequency of words in the dataset.

This work is dedicated first and foremost to my father Jamal Sakakini (I'm sure he would like to see his name) and my mother (she actually doesn't care), for all the sacrifices they have made, each in their own way. It is actions like that which remind us of the importance of the family institution in a world full of atrocities. I would also dedicate this work to my grandfathers and grandmothers. Unconditional love can do magic. This work is also dedicated to my oldest brother Mohammad for his mentorship in life, together we have made great life decisions. As for my older brother Abdul Rahman, your company in life, kept me happy and balanced to be where I am today. Finally I direct my thanks to my friends Charbel, Pamela, Patrick, and Peter for a beautiful life outside work.

ACKNOWLEDGMENTS

This work would have not seen life were it not for the guidance of my advisers: Pramod Viswanath and Suma Bhat. The fruitful discussions challenging the current trends in natural language processing have paved the way for this work. Fruitful discussions with my partners in NLP, Jiaqi Mu and Hongyu Gong, have been also a great help.

I would also like to thank Paritosh Garg for his technical assistance in using heavy machinery for this computationally demanding research.

My thanks are also extended to the senior in the group, my mentor Peter Kairouz, for passing along his successful PhD experience to me.

TABLE OF CONTENTS

LIST OF TABLES vi

LIST OF FIGURES vii

LIST OF ABBREVIATIONS viii

CHAPTER 1 INTRODUCTION 1
1.1 Problem Statement and Motivation 1
1.2 Related Work 2
1.3 Contributions and Results 3

CHAPTER 2 BACKGROUND 5
2.1 Word Representations 5
2.2 Unsupervised Morphology Induction 7

CHAPTER 3 FRAMEWORK 9
3.1 Automatic Generation of Analogy Datasets 10

CHAPTER 4 EXPERIMENTS 11
4.1 Unsupervised Morphology Extraction 11
4.2 Study 1: Validating the Generated Dataset 11
4.3 Study 2: Customization Capabilities of our Method 13
4.4 Study 3: Dataset Generation for Other Languages 14
4.5 Study 4: Reformulating the Analogy Question 15

CHAPTER 5 RESULTS 16
5.1 Study 1: Validating the Generated Dataset 16
5.2 Study 2: Customization Capabilities of our Method 16
5.3 Study 3: Dataset Generation for Other Languages 20
5.4 Study 4: Reformulating the Analogy Question 20
5.5 General Discussion 21

CHAPTER 6 CONCLUSION AND FUTURE WORK 25

REFERENCES 26

LIST OF TABLES

2.1 Top rules in terms of hit rate 8

4.1 Equivalent surface form rules of the Google dataset's sections. 12
4.2 Select rules for every language considered 14

5.1 Accuracy values of different word representations on both the Google dataset and the generated dataset. The dimension size is represented by n. 18
5.2 Comparison between the analogy instances of the dataset with the most frequent words (average frequency = 353549.4) and the dataset with the least frequent words (average frequency = 7117.3) 18
5.3 Accuracy of Polyglot word representations on generated analogy task datasets of respective language 20
5.4 Accuracy of pre-trained Word2Vec representations on both datasets before and after reformulation of the analogy question. 20
5.5 A sample of representative word pairs (column 2) - those *closest* to the centroid - encoding the relation (morphological rule) in column 1. As a comparison, the word pairs in column 4 are *farthest* from the centroid. 21
5.6 Manual evaluation of the correctness of instances in the generated dataset per section. 22
5.7 Examples of the effect of semantic similarity between the two sides of an analogy question on the distance of the analogy. The right (left) column shows the top (bottom) analogy questions in terms of cosine similarity. The rule considered is the comparative rule. 23

LIST OF FIGURES

5.1 Performance of different word representations on both datasets. Representations are ordered as in Table 5.1. 17
5.2 Performance of different word representations on datasets of different word-frequency properties 17
5.3 Histogram of the distances of word pairs from the centroid to show the varying degrees to which word pairs encode a relation (left: adjective-adverb with suffix -ly, right: antonym with prefix un-) 21
5.4 Impact of semantics on the analogy question in the comparative section 24

LIST OF ABBREVIATIONS

Ar	Arabic
CBOW	Continuous Bag of Words
CWin	CWindow
De	German
En	English
Es	Spanish
Fr	French
GloVe	Global Vectors
GPU	Graphical Processing Unit
GRE	Graduate Record Examination
He	Hebrew
MAT	Miller Analogies Test
MSR	Microsoft Research
NLP	Natural Language Processing
PMI	Pointwise Mutual Information
POS	Part of Speech
PPMI	Positive Pointwise Mutual Information
SG	Skip-Gram
SSG	Structured Skip-Gram
SVD	Singular Value Decomposition

CHAPTER 1

INTRODUCTION

1.1 Problem Statement and Motivation

The field of natural language processing (NLP) came to existence with the purpose of equipping machines with the capability of understanding and generating natural language. One of the basic units of natural language is words, and hence to understand or generate pieces of text, machines need to first grasp the meaning of what composes text: words.

With this in mind, comes the critical problem of word representation. By word representation, we do not mean representation of words at the surface level (i.e., specification of the characters that make up the word), but we actually mean representation of words at the semantic level. Such representations are recently gaining momentum in comparison to surface level representations. This is due to the fact that a deeper understanding of the word leads to smarter machines.

With the rise of such representations and a huge variety of them, it is in the NLP community's interest to faithfully evaluate such representations. Faithful evaluation of word representation algorithms can lead to the understanding of the downfalls of current algorithms as well as insights into opportunities of enhancement. Moreover, faithful evaluation methods can direct the NLP community to the best practices of word representations when put in use in the building of larger NLP systems.

1.2 Related Work

Distributed word representations have been shown to represent meaning via geometry, i.e. the linguistic relationship between two words is captured via the *difference* between their corresponding vectors [1]. How good the word representations are is typically evaluated either via extrinsic or intrinsic evaluations. In an extrinsic evaluation, the metric considered is the performance of a full-fledged downstream system with word representations as an input (examples: sentiment analysis [2], parsing [3, 4], chunking [5], and named entity recognition [6]). Intrinsic evaluation directly tests for attributes desired of the word representations broadly classified into four categories: *relatedness*, *analogy*, *categorization*, and *selectional preference* [7].

This research focuses on the analogy task as an intrinsic evaluation method. A word analogy task involves predicting w_d that fits w_a to w_b as w_c to w_d, i.e., $w_a : w_b :: w_c : w_d$. This assumes w_a and w_b to be linked by a semantic relation and expects the same relation to link w_c and w_d. Such analogy tasks (also termed proportional analogies) are important metrics of language understanding in psychometric tests, such as the Miller Analogies Test (MAT) and the Graduate Record Examination (GRE).

In [1], word representations were found to exhibit semantic regularities: words sharing a similar relationship exhibit equivalent difference vectors, with the difference vector encoding the semantic relation. An example of a semantic analogy question is, given *boy* is to *girl* as *brother* is to x, the word representations are able to guess that x is *sister* through linear operations. An example of a syntactic analogy task is, given *hopeful* is to *hopefully* as *quick* is to x, the word representations are able to guess that x is *quickly*, also through the same operation. The operation performed on the word representations to predict the word fitting the semantic relation is the following: $w_d = \arg\max_{x \in V} cos(\vec{x}, \vec{w_b} - \vec{w_a} + \vec{w_c})$. The accuracy of word representations at guessing w_4 is the evaluation metric used in the analogy task and the types of semantic relations between word pairs range from those encoded with the use of morpho-syntactic markers (to indicate that the words are related with respect to the grammatical category of number or degree of comparison) to those purely lexicalized (such as currency of country or capital of country).

Further, the efficacy of word representations in solving the analogy task was demonstrated via empirical performance over two hand-crafted datasets: [1] (referred to as the Google dataset in the rest of this document) and the MSR dataset [8]. Both the datasets include word pair instances collected with significant dependence on linguistic tools and resources (examples: POS tagger and semantic relation similarity tools [9] in the Google data set and knowledge from Wikipedia, a dictionary, and WordNet in [8]). The same reliance on linguistic tools and resources in conjunction with a human annotation continues in the creation of other datasets (meant to handle categories other than analogy): relatedness [10–19], categorization [20–22], and selectional preference [23, 24]. To summarize: a key feature in the creation of evaluation datasets is the extensive need for *manual labor* and *linguistic tools/resources*, borrowing *opportunistically* from disparate studies.

Other works have aimed at creating new evaluation metrics. The work in [25] claims to have developed the only intrinsic evaluation metric for word representations. Their method relies on a POS tagged corpus which is used to create "gold" word representations. Subspace alignment is performed between the evaluated and the gold representations, and the correlation metric reflects the "goodness" of the evaluated word representations. The disadvantage of such a method is its limit to languages high on resources (specifically high quality POS taggers), besides relying on a "gold" set of representations which are not proven to be truly the "gold". On the other hand, the work in [7] devises a method to compare word representations instead of evaluating each absolutely. The method uses crowd sourcing tools to request humans' help in evaluating word representation at retrieving the most similar word. The disadvantage of this method is the need of human input for every comparison that needs to be made.

1.3 Contributions and Results

The main contributions of this research are:

- A method to generate more reliable syntactic analogy datasets in an automatic and unsupervised fashion and that can be applied to any language.

- The capability to generate custom datasets based on frequency and POS tag preferences.
- An extension of the method in [26] to detect infix rules.
- The creation of the first analogy datasets for French, Spanish, German, Arabic and Hebrew.
- A concrete reformulation of the analogy task whereby the analogy is specified by all instances of that analogy instead of an arbitrary one, leading to enhanced performance of word representations on the analogy task.
- A critical observation on the effect of semantic relatedness on the analogy task, and how our reformulation effectively eliminates that effect.

A 95% Pearson correlation coefficient on the performances of different word representations on the our generated dataset with those on the google dataset reflects the ability of our algorithm to generate datasets of comparable quality.

CHAPTER 2

BACKGROUND

2.1 Word Representations

Traditionally, a word in an NLP system is represented via a one-hot vector. In a vocabulary V, taking a word $w \in V$, and assuming i is the index of w in V, $\vec{w}$ is set to all zeros, with one in the index corresponding to the word at hand, i.e. $\vec{w_t} = 0, \forall t \neq i$, and $\vec{w_i} = 1$. Such a representation sets all words in the vocabulary at an equal distance in the space $\mathbb{R}^{|V|}$, although semantically speaking, “book” is closer to “textbook” than “car”. Because of that, an NLP system could not statistically learn about “textbook”, for example, from learning about the word “book”.

To overcome this shortcoming in one-hot vector representation of words, researchers in the field resorted to distributed representation of words. This family of representations bases its core on the distributional theory of [27] that states that similar words keep the same company of words.

2.1.1 PPMI-based

Starting from the previously mentioned distributional theory of words, the aim is to represent a word through how much it associates with every other word. For that we rely on counting how much a certain word (c) appears in the context of the word (w) under consideration. To evaluate the association between the two words we resort to the pointwise mutual information metric (PMI). The PMI is estimated as follows:

$$PMI(w,c) = log(\frac{\hat{P}(w,c)}{\hat{P}(w) * \hat{P}(c)}) = log(\frac{\frac{count(w,c)}{|D|}}{\frac{count(w)}{|D|} * \frac{count(w)}{|D|}})$$

In the previous equation, count(w,c) refers to the count of occurrences of c in the context of w. Context could be defined in different ways. Context is usually defined as within a window of k words from w. Other definitions could consider the distance in a syntactic dependency tree instead. The term $|D|$ refers to the number of tokens in the document.

To avoid values of negative infinity when count(w,c)=0, and to avoid negative values in general, researchers have resorted to positive pointwise mutual information (PPMI) instead. It was shown in [28] that PPMI-based word representations outperformed PMI-based representations.

2.1.2 SVD

Word representations in the PPMI regime are high dimensional and sparse. Dimensionality is $|V|$ in our case, and O($|V|$) in general. For computational purposes, it was in the interest of researchers to create more dense and less dimensional word representations. For that they resorted to the standard dimensionality reduction method, namely truncated singular value decomposition (SVD).

Taking M^{PPMI} to be the matrix that holds the PPMI-based word representations (every row corresponds to the representation of one word in the vocabulary), we perform SVD on it.

$$M^{PPMI} = U \cdot \Sigma \cdot Y^T$$

where U is $|V| \times r$, Σ is $r \times r$, Y is $|V| \times r$, and r is the rank of M^{PPMI}. Assuming that the desired low dimensionality is of size d, we would take the first d columns of U and the top d eigenvalues in Σ. The matrix holding our word representations would become:

$$W^{SVD} = U_d \cdot \Sigma_d$$

where W^{SVD} is a $|V| \times d$ matrix instead of $|V| \times |V|$, with $d <<< |V|$.

2.2 Unsupervised Morphology Induction

It was shown in [1], that these distributed word representations encode information on morphology. For example, you would expect to find the difference vector $\vec{v}_{horses} - \vec{v}_{horse}$ close to the difference vector $\vec{v}_{cows} - \vec{v}_{cow}$. Exploiting this regularity in the representation, the authors in [26] devise a method for unsupervised morphology induction. As described in more detail in [26], the algorithm divides into four steps:

2.2.1 Candidate Rule Extraction

Morphological rules, in general, are translated orthographically through affixing. For this reason, candidate rules are extracted by checking for every $(w_1, w_2) \in V^2$ possible suffix, prefix substitutions from w_1 to w_2. A rule is represented as type:seq1:seq2, where type $\in$ {pre,suf}, seq1 is the sequence of characters to be removed from w_1 and seq2 is the sequence of characters to be added to reach w_2. For every candidate rule r, there exists a support set S_r such that:

$$S_r = \{(w_1, w_2) \in V^2 | w_1 \xrightarrow{r} w_2\}.$$

2.2.2 Word Representations Generation

The algorithm authenticates candidate morphological rules if the difference between the words in the support set shows a systematic pattern in the vector space. Hence word representations in the vector space are needed. Any off-the-shelf algorithm for word representation generation could be used.

2.2.3 Candidate Rules Authentication

To check for the authenticity of a rule r, we check for the percentage of $S_r \times S_r$ for which a linear relationship exists in the vector space. For that we define the notion of a hit: $hit = \mathbb{1}(cos(\vec{w_2}, \vec{w_4} - \vec{w_3} + \vec{w_1}) \geq t_{sim})$, where $(w_1, w_2), (w_3, w_4) \in S_r$ and $0 \leq t_{sim} \leq 1$ stands for cosine similarity threshold. Moreover, a hit rate of a specific rule r is defined as follows:

Table 2.1: Top rules in terms of hit rate

rule	**hit rate**	**Example**
$suf : sed : zed$	100	organised
$suf : sed : ze$	93.91	organised
$suf : y : ical$	90.35	history
$suf : m : tic$	78.69	tourism
$suf : y : ies$	73.42	penny

hit rate $= \frac{\text{hit count}}{|S_r \text{x} S_r|}$. A rule is authenticated if its hit rate is above a threshold t_{hr}. Table 2.1 shows the top rules extracted in terms of hit rate.

2.2.4 Morphological Rule Disambiguation

The authentication of a rule does not imply that all the instances in its support set are valid instances. Take for example the rule $suf : \epsilon : ly$. The rule is valid and expected to be authenticated but (on, only) is an invalid instance of its support set.

Moreover, one orthographic rule could correspond to multiple morphological rules. Take, for example, the rule $suf : \epsilon : s$. The surface transformation represents the plural rule (player, players) and it also represents the person case change rule (think, thinks). To tackle these two issues, the following algorithm is proposed.

We greedily search for the instance $(w_1, w_2) \in S_r$ that "hits" with most of the instances in S_r. These instances form $S_r^{(w_1,w_2)}$. Then we iterate on $S_r - S_r^{(w_1,w_2)}$ until the largest possible subset to form is below a threshold in size.

CHAPTER 3

FRAMEWORK

To automatically generate a syntactic analogy dataset we follow an unsupervised morphology induction approach to infer: **(1)** the morphological rules of a language, **(2)** within a morphological rule, the knowledge of pairs of words adhering to that rule [26].

The system needs as an input only one item: a monolingual corpus. This reflects the system's ability to be more widely applicable to other languages, even low-resource ones. For a detailed description of our implementation of the unsupervised morphology induction system of [26], the reader is directed to Section 2.2.

To cover a wider range of languages and respective rules, we expand the morphology induction system in [26], limited to prefix/suffix rules, with a heuristic to detect infix rules. For brevity, we only show how to detect candidate infix rules while mentioning that the rest of the system is inline with the method of [26].

Infix rules in Semitic languages exhibit two properties [29]: (**1**) root words are dominantly triliteral, (**2**) the three letters of the root word appear in the same order in the derived word. Hence, for all (w_1, w_2) $\in V^2$, if length(w_1) $= 3$ and the letters of w_1 keep positional order in w_2, then (w_1, w_2) is added to the support set of the rule (inf: $r_1r_2r_3$: $s_1r_1s_2r_2s_3r_3s_4$). In the naming of the infix rule, r_1, r_2, r_3 abstract the specific letters of the root, and s_1, s_2, s_3, s_4 indicate the strings of letters that come in between the root letters to form the derived word.

3.1 Automatic Generation of Analogy Datasets

The result of the unsupervised morphology induction stage is a set of morphological rules for the language, and pairs of words adhering to each rule. It should be noted that the support sets of some rules are unsupervisedly divided into subsets due to having one surface form rule correspond to multiple morphological rules, as mentioned in [26].

A high hit rate indicates valid morphological rules, and the support set of those morphological rules indicates the correct pairs of words adhering to the respective morphological rule. Hence, for a given number of rules desired (k) and the number of instances conforming to each rule $\{s_1, s_2, ..., s_k\}$, the algorithm to create a syntactic analogy dataset is as follows:

1. Select the top k rules in terms of hit rate to form the set R (in case of a priori knowledge of morphological rules in a language, rules could be manually specified instead).
2. For every rule $r_i \in$ R, select $S_{r_i}^{(w_1,w_2)} \subset S_{r_i}$ with the largest cardinality.
3. Downsample $S_{r_i}^{(w_1,w_2)}$ to size s_i to control the size of the section in the dataset.
4. Create the syntactic analogy section of size $s_i.(s_i - 1)$ for rule r from $S_{r_i}^{(w_1,w_2)}$x$S_{r_i}^{(w_1,w_2)}$

CHAPTER 4

EXPERIMENTS

We divide our experiments into four parts. In the first, we first empirically validate our generated dataset. In the second, we show experiments on the customization capabilities of our method. The third part demonstrates the quality of our method on other languages. Finally, the fourth part checks for the impact of the reformulation of the analogy question.

4.1 Unsupervised Morphology Extraction

A prerequisite to all of our experiments is the unsupervised learning of morphological rules as described in Section 2.2. We use off-the-shelf word embeddings, trained on the English Wikipedia and limited to the top 100K words in terms of frequency and made available in Polyglot [30]. These word representations showed state-of-the-art results when used in a part-of-speech tagging task. A cosine similarity threshold $t_{sim} = 0.5$ is used.

4.2 Study 1: Validating the Generated Dataset

Given that all the instances in the Google dataset are correct analogies, we take it as the gold dataset. To generate the same sections as the Google dataset we transfer the section rules to their surface form equivalent to the extent possible. Table 4.1 shows the surface form rules selected as equivalent to Google's sections. We omit gram6-nationality-adjective since it is not a morphological alternation. Also notice that gram8 and gram9 map to the same rule and thus only the rule with the larger support set would be represented as indicated by the second step in our method.

Table 4.1: Equivalent surface form rules of the Google dataset's sections.

Google section	Generated section
gram1-adjective-to-adverb	$suf : \epsilon : ly$
gram2-opposite	$pre : \epsilon : un$
gram3-comparative	$suf : \epsilon : er$
gram4-superlative	$suf : \epsilon : est$
gram5-present-participle	$suf : \epsilon : ing$
gram7-past-tense	$suf : ing : ed$
gram8-plural	$suf : \epsilon : s$
gram9-plural-verbs	$suf : \epsilon : s$

To prove that the generated dataset achieves the same quality of comparative evaluation as the gold dataset, we check the correlation of performance scores of multiple word representations on the generated dataset with the performance scores of the same word representations on the Google dataset. By comparative evaluation, we mean the dataset's ability to rank word representations and reflect the difference in their quality.

We use five different word representations for this experiment. Each one is trained on the same English Wikipedia dump (as of 01/15/2016). The methods for generating the five word representations are:

- **CBOW and Skip-Gram (SG):** These two methods were introduced in [31] under the Word2Vec family, a neural-network-based method.
- **CWindow (CWin) and Structured Skip-Gram (SSG):** These two methods are syntactic modifications to the two methods in Word2Vec. They were introduced in [32].
- **GloVe:** This one is under a different paradigm of word representations. Word representations are constructed from word co-occurrences. This method was introduced in [33].

Each word representation was trained with five different dimension sizes (100, 200, 300, 400, 500), yielding 25 different word representations in all. It is important to note that this study is concerned with the evaluation methods for word representations rather than comparing different methods for creating word representations. The word representations were trained under default settings and not optimized for the purpose of comparison.

4.3 Study 2: Customization Capabilities of our Method

In addition to automatically generating analogy sections of a dataset, we provide the ability to customize the selection as well. This helps the research community by giving a deeper insight in evaluating word representations. In particular, in this study we provide frequency-related, morphological rule-related and size-related customization handles.

Frequency: In this experiment, we study the algorithm's ability to generate datasets of different frequency properties. This demonstrates the effects brought about by such a frequency variation, which can potentially offer valuable insights to researchers. Towards this, we show the variation of performance of different word representations on generated datasets of different frequency properties.

We achieve this variation as follows:

- generate all possible analogy instances using the support set of a rule r;
- order these instances in terms of average frequency of the four words included in the analogy;
- divide them into five different equal bands;
- sample 3000 instances from these bands.

This results in five datasets of the following average word frequency values: f1=7117.3, f2=15596.4, f3=30050.1, f4=61737.3, f5=353549.4. We average to account equally for the quality of all word representations participating in the analogy. These values are the average of the frequency values of all the words in the analogy dataset. Frequency values are extracted from the Wikipedia dump the word representations were trained on.

Morphological Rules: A second tunable aspect provided by our method is the choice of different morphological rules so as to evaluate the performance of different word representations on them. The Google dataset offers nine rules, which is a small portion of the possible morphological transformations in English. Moreover, the control over the rules is equivalent to control over the POS tags, which has been of interest, as in [7].

Table 4.2: Select rules for every language considered

French	Spanish	German
$suf : ont : a$	$suf : mos : ron$	$suf : en : t$
$suf : ons : ent$	$suf : ron : ban$	$suf : \epsilon : n$
$suf : \epsilon : s$	$suf : \acute{o} : aron$	$suf : s : n$
$suf : ant : \acute{e}$	$suf : n : ron$	$suf : st : t$
$suf : ons : ez$	$suf : \acute{a}n : on$	$suf : \epsilon : es$
$suf : \epsilon : e$	$suf : z : ces$	$suf : s : m$
$suf : s : es$	$suf : ra : ndo$	$suf : s : r$
$pre : \epsilon : ir$	$suf : os : as$	$suf : \epsilon : st$

Size: A third benefit of our algorithm is its ability to give the researcher control over the size of every section. Increasing the size of a section makes the evaluation more reliable. Increasing the size does increase the evaluation time, but the evaluation algorithm is fully parallelizable. Hence, this issue could be alleviated with the use of GPUs.

4.4 Study 3: Dataset Generation for Other Languages

Our next study highlights the fundamental contribution of our work to the NLP community in general and word representations in particular. We exploit the language agnostic property of our algorithm presented to generate datasets for languages other than English using their monolingual corpora alone as input. In this experiment the respective Wikipedia repositories were used to create the needed word representations available in [30].

This being a proof-of-concept, we select the rules that encode tense, aspect, gender and number in French, German and Spanish to create analogous datasets. Table 4.2 shows a subset of the rules considered for every language. As for Arabic and Hebrew, we use complete unsupervision. The system selects the top 8 rules in terms of hit rate rather than us supplying the rules. For a detailed look at the rules selected and the instances generated, the datasets are published online. [1]

[1] https://uofi.box.com/s/vla63rquqdpwo0k9uipkfjug493alxqy

4.5 Study 4: Reformulating the Analogy Question

In the current formulation of the analogy question ($w_a : w_b :: w_c : w_d$), one pair of words ($w_a : w_b$) defines the analogy. We hypothesize that this is an underspecification of the analogy relation in the vector space. In the new formulation we replace $\vec{w_b} - \vec{w_a}$ by $\underset{(w_1, w_2) \in S_r}{\text{average}} (\vec{w_2} - \vec{w_1})$ for every rule r in the dataset.[2]

A byproduct of this reformulation is the reduction of the size of the analogy dataset from n^2 to n, where n is the number of unique pairs of words in the analogy questions of a dataset. Thus, another benefit of this reformulation is the computational speedup while keeping the number of unique pairs of words constant.

The experiment performed checks for the change in the performance of word representations after the reformulation of the analogy task. For the purpose of this experiment we use pre-trained English word representations made available online.[3] These 300-dimensional word representations were trained on part of the Google News dataset ($\approx$ 100 billion words).

[2] For the purpose of the immediate use of our published datasets, we represent the average difference vector by the difference vector of the pair closest to it.

[3] https://code.google.com/archive/p/word2vec/

CHAPTER 5

RESULTS

5.1 Study 1: Validating the Generated Dataset

To show that the generated dataset is comparable in quality with the Google dataset, we use the 25 word representations and obtain their corresponding accuracy values on the generated and the Google datasets, yielding 25 comparison pairs of data points, plotted in Figure 5.1. We observe that the plot of accuracy values on the generated dataset is a mere shifted and scaled version of the plot of accuracy values on the Google dataset. This reflects the high correlation between both sets of accuracy values, thus confirming the comparable quality of both datasets. We confirm this observation using a more objective quantifier, the Pearson correlation coefficient r between the two sets of accuracy values. A high value of $r = 0.95$ validates our results.

A detailed view of the 25 pairs of data points is shown in Table 5.1.

5.2 Study 2: Customization Capabilities of our Method

As mentioned in Section 4.3, one of the benefits of the automatic generation of analogy datasets is the ability to customize the dataset. The properties considered here are: Frequency, Morphological Rules and Size. Following are the results of the experiments to demonstrate the algorithm's customizability as well as the insights it could bring in evaluating word representations.

Frequency: To demonstrate the algorithm's ability to accommodate different word-frequency properties, we create five different datasets of respective average word frequency values: 7117.3, 15596.4, 30050.1, 61737.3, 353549.4

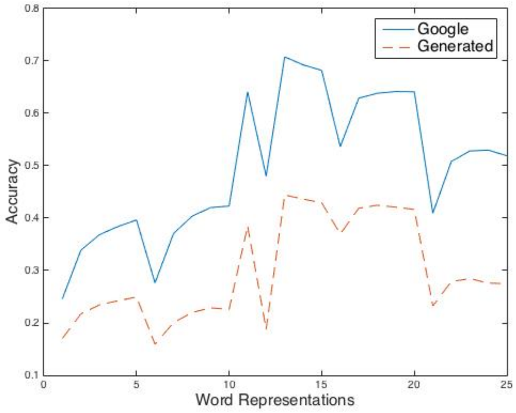

Figure 5.1: Performance of different word representations on both datasets. Representations are ordered as in Table 5.1.

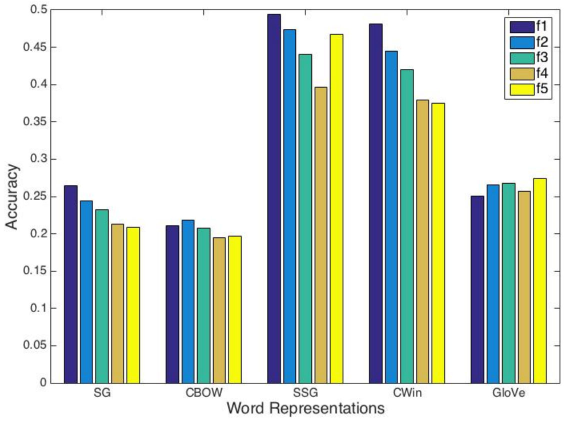

Figure 5.2: Performance of different word representations on datasets of different word-frequency properties

Table 5.1: Accuracy values of different word representations on both the Google dataset and the generated dataset. The dimension size is represented by n.

Method	n	Google	Generated
SG	100	24.63	17.04
	200	33.89	21.73
	300	36.85	23.46
	400	38.40	24.24
	500	39.66	24.95
CBOW	100	27.64	15.95
	200	37.08	20.05
	300	40.37	22.00
	400	42.03	22.87
	500	42.31	22.59
CWin	100	64.10	38.48
	200	48.02	18.80
	300	70.80	44.42
	400	69.26	43.64
	500	68.20	42.93
SSG	100	53.64	36.98
	200	62.90	41.89
	300	63.84	42.47
	400	64.18	42.09
	500	64.13	41.66
GloVe	100	40.95	23.24
	200	50.83	27.91
	300	52.83	28.46
	400	52.98	27.60
	500	51.90	27.50

Table 5.2: Comparison between the analogy instances of the dataset with the most frequent words (average frequency = 353549.4) and the dataset with the least frequent words (average frequency = 7117.3)

Most Frequent				**Least Frequent**			
most	mostly	new	newly	diligent	diligently	seeming	seemingly
built	unbuilt	official	unofficial	restrained	unrestrained	paved	unpaved
play	player	work	worker	import	importer	ranch	rancher
high	highest	old	oldest	bright	brightest	quick	quickest
march	marching	work	working	modify	modifying	cull	culling

as detailed in Section 4.3. The ability to create these datasets indicates the algorithm's ability to control the word-frequency property of the dataset.

For a closer look at the instances from different frequency bands, we refer the reader to Table 5.2, which considers two datasets: the one with the lowest frequency property ($f1$), and the one with the highest frequency property ($f5$).

After demonstrating the algorithm's ability to customize the words in the generated dataset by frequency, we show what kind of insights it could bring when evaluating word representations. Hence, we evaluate the performance of the five trained word representations (of dimension 300) on the five different datasets. Figure 5.2 shows the performance results for every set of word representations. We point out that (SG, SSG, CWin) actually perform better on datasets with less frequent words, whereas CBOW and GloVe maintain a more consistent performance across. Although the results are counter-intuitive, we note that the experiments are performed on the top 100K words in terms of frequency, which are all expected to be well represented in the vector space.

This shows how representations are differentially susceptible to the frequency of words, but we refrain from explaining the reason for the phenomenon (such as method or hyperparameter selection).

Morphological Rules & Size: Study 1 showed our method's ability to construct quality datasets based on predefined rules. Since the generation method is not specific to the rules selected, any morphological rule could be added to the dataset.

To put the algorithm's ability to enlarge the dataset into perspective, the section on adjective-adverb in the Google dataset is of size 992. Our algorithm can generate a section on this rule of size $710 * 709 = 503390$ and this increase directly impacts the reliability of the performance measures.

Table 5.3: Accuracy of Polyglot word representations on generated analogy task datasets of respective language

Fr	Es	De	En	Ar	He
17.10	17.22	15.35	14.64	13.02	12.15

Table 5.4: Accuracy of pre-trained Word2Vec representations on both datasets before and after reformulation of the analogy question.

	Google	Generated
Before Reformulation	71.92	47.29
After Reformulation	75.28	61.34

5.3 Study 3: Dataset Generation for Other Languages

We create analogy datasets for French, Spanish, German, Arabic, and Hebrew. Next, we evaluate the word representations in these languages made available in Polyglot [30] on the respective analogy dataset. The accuracy scores are indicated in Table 5.3. The scores range between 12% and 18%, where the analogous word representation and generated dataset on English return an accuracy of 15%. This indicates that the generated datasets are suitable for the analogy task and are of comparable quality to that in English. The low performance of the Polyglot word representations is largely due to the small dimension size of 64.

5.4 Study 4: Reformulating the Analogy Question

To validate the benefit of the reformulation of the analogy question, we check the change in the performance of pre-trained Word2Vec representations on both datasets (generated and Google) after reformulation. The results of this experiment are shown in Table 5.4. The increase in accuracy values after reformulation validates our hypothesis that the reformulation better represents the analogy task for word representations. We highlight that the gains are even more significant for the generated dataset where robustness is more called for, given the noise in the dataset.

As a result of the reformulation, we notice that not all word pairs conforming to a relation ‘encode’ the relation to the same extent; some are more canonical than others as seen in the histograms of the distance from the cen-

Table 5.5: A sample of representative word pairs (column 2) - those *closest* to the centroid - encoding the relation (morphological rule) in column 1. As a comparison, the word pairs in column 4 are *farthest* from the centroid.

Rule	**Closest**	Distance	**Farthest**	Distance
suf:ϵ:ly	(random,randomly)	0.76	(amp,amply)	-0.02
pre:ϵ:un	(fortunately,unfortunately)	0.77	(commonly,uncommmonly)	0.05

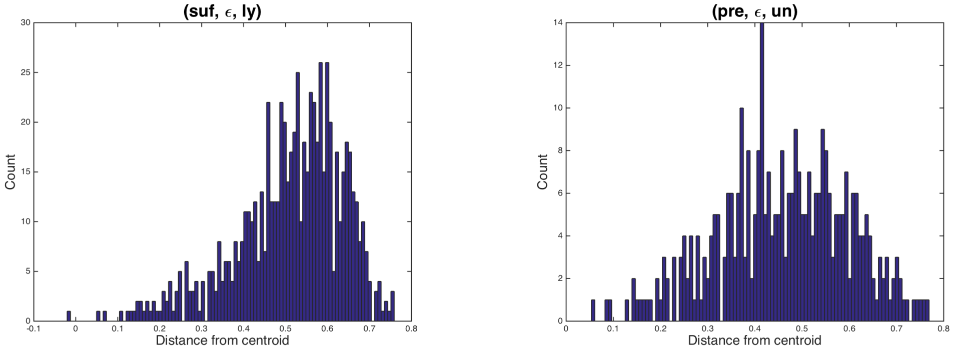

Figure 5.3: Histogram of the distances of word pairs from the centroid to show the varying degrees to which word pairs encode a relation (left: adjective-adverb with suffix -ly, right: antonym with prefix un-)

troid, Figure 5.3, for the adjective-adverb relation and the antonym using the prefix un-. We also tabulate the representative word pairs for the semantic relations encoded in the stems of the analogy in Table 5.5. This suggests that the degree to which a word pair encodes a given relation could be chosen as part of the dataset creation.

5.5 General Discussion

Although Figure 5.1 and a Pearson correlation coefficient of 0.95 are sufficient to infer the generated dataset's ability to compare word representations, we acknowledge that it cannot be used as an absolute metric for evaluating the word representations on the analogy task. In fact, the accuracy values on the generated dataset are an underestimate as shown in Figure 5.1. While this can be attributed to the noisy instances in the generated dataset, the noise does not affect the comparative evaluation, as was evident from the results in Figure 5.1, since any noise is expected to affect all word representations equally.

Table 5.6: Manual evaluation of the correctness of instances in the generated dataset per section.

rule	type	count
$suf : \epsilon : est$	valid	41
	invalid	2
$suf : \epsilon : ly$	valid	60
	invalid	0
$suf : \epsilon : ing$	valid	60
	invalid	0
$suf : ing : ed$	valid	60
	invalid	0
$pre : \epsilon : un$	valid	60
	invalid	0
$suf : \epsilon : s$	noun	44
	verb	7
	both	9
$suf : \epsilon : er$	comparative	15
	verb-nominal	41
	invalid	3

Statistically speaking, the probability for a noisy analogy question to affect all n different word representations equally is $P(\text{equal effect}) = \frac{(|V|-1)^n}{|V|^n}$, where $|V|$ is the vocabulary size and equal to 100K in our experiments.

Delving deeper into the reason behind the introduction of noisy instances in the generated dataset, we manually evaluate the distribution of instances in every section and show the results in Table 5.6. We notice that only 5 out of 402 are invalid instances of morphological rules. The pair (*dens, densest*), where *dens* is the plural of *den*, was wrongly identified as an instance of the superlative rule $suf : \epsilon : est$. The rest of the noise is due to the ambiguity of the surface form of the rule. For example, $suf : \epsilon : er$ could represent the comparative rule (fast, faster) as well as the verb-nominal rule (play, player). We point out that some of the instances in the $suf : \epsilon : s$ belong to both the noun category as well as the verb category, e.g., (walk, walks). This ambiguity in the surface form results in noisy instances like the following: $player : players :: generalize : generalizes$.

In this context, we would like to note that the method for comparative evalu-

Table 5.7: Examples of the effect of semantic similarity between the two sides of an analogy question on the distance of the analogy. The right (left) column shows the top (bottom) analogy questions in terms of cosine similarity. The rule considered is the comparative rule.

Top Analogy Questions				**Bottom Analogy Questions**			
large	larger	*big*	bigger	bad	worse	new	newer
strong	stronger	*hard*	harder	low	lower	old	older
tight	tighter	*tough*	tougher	cool	cooler	old	older
hard	harder	*tough*	tougher	great	greater	old	older
big	bigger	*large*	larger	new	newer	great	greater

ation mentioned in [7] requires costly human evaluation, whereas our method requires no supervision.

Finally, we make an important observation on the effect of semantic relatedness on the analogy task: The more semantically related the base forms of both sides of the analogy question are, the “easier” the analogy question is (i.e. the answer is closer to the linear combination of the three input words). Figure 5.4 shows the correlation between semantic relatedness and the analogy task through the high cosine similarity between the base forms of the top 500 analogy questions in terms of “easiness”. Taking a specific example, the top instance in terms of analogy distance is $large : larger :: big : bigger$ (more examples are shown in Table 5.7). This is another argument in favor of our reformulation of the analogy question, since this issue does not exist after reformulation.

Finally, one could suspect that the word representations used for unsupervised morphology induction have a comparative advantage over others. Such a claim was empirically proven wrong by adding English Polyglot representations (which were used for the automatic generation of the English dataset) to the calculations of r done over the results in Table 5.1, increasing it from 0.953 to 0.956.

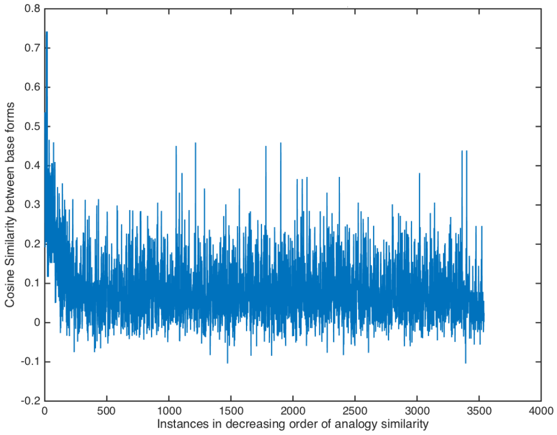

Figure 5.4: Impact of semantics on the analogy question in the comparative section

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this research we presented a method to automatically generate analogy task datasets to evaluate word representations. This automatic generation allows for customization based on researchers' needs and required insights, with a provision to increase the size of datasets for a more reliable evaluation. The language-agnostic property of our method allows it to be applied to any language, providing the ability to create datasets for languages other than English where evaluation benchmarks are limited if not nonexistent. A high correlation with the evaluation on the Google dataset reflects the quality of the method used in the automatic generation of the analogy task datasets.

To improve the generated dataset's capability of absolute evaluation we plan to remove the noise through enhancing the methods used to disambiguate surface form rules.

Although the accuracy values on the analogy task are not of central interest in this study, we acknowledge that the evaluation on the generated datasets is an underestimate of the word representations' ability to solve the analogy task. Given that the primary source of noise is the ambiguity of the rules, we are currently investigating suggested ways to enhance the disambiguation capability of the unsupervised morphology induction framework by using rank-based ordering instead of cosine similarity based ordering of hit rates, as well as using better quality word representations to learn the morphological rules.

REFERENCES

[1] T. Mikolov, W.-T. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations." in *HLT-NAACL*, 2013, pp. 746–751.

[2] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, vol. 1631, 2013, p. 1642.

[3] M. Bansal, K. Gimpel, and K. Livescu, "Tailoring continuous word representations for dependency parsing." in *ACL (2)*, 2014, pp. 809–815.

[4] A. Lazaridou, E. M. Vecchi, and M. Baroni, "Fish transporters and miracle homes: How compositional distributional semantics can help NP parsing," in *Proc. of EMNLP*, 2013.

[5] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics, 2010, pp. 384–394.

[6] J. Guo, W. Che, H. Wang, and T. Liu, "Revisiting embedding features for simple semi-supervised learning," in *EMNLP*, 2014, pp. 110–120.

[7] T. Schnabel, I. Labutov, D. Mimno, and T. Joachims, "Evaluation methods for unsupervised word embeddings," in *Proc. of EMNLP*, 2015.

[8] B. Gao, J. Bian, and T.-Y. Liu, "Wordrep: A benchmark for research on learning word representations," arXiv preprint arXiv:1407.1640, 2014.

[9] D. A. Jurgens, P. D. Turney, S. M. Mohammad, and K. J. Holyoak, "Semeval-2012 task 2: Measuring degrees of relational similarity," in *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1.* Association for Computational Linguistics, 2012, pp. 356–364.

[10] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," in *Proceedings of the 10th International Conference on World Wide Web.* ACM, 2001, pp. 406–414.

[11] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa, "A study on similarity and relatedness using distributional and wordnet-based approaches," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics.* Association for Computational Linguistics, 2009, pp. 19–27.

[12] G. A. Miller and W. G. Charles, "Contextual correlates of semantic similarity," *Language and Cognitive Processes*, vol. 6, no. 1, pp. 1–28, 1991. [Online]. Available: http://dx.doi.org/10.1080/01690969108406936

[13] H. Rubenstein and J. B. Goodenough, "Contextual correlates of synonymy," *Commun. ACM*, vol. 8, no. 10, pp. 627–633, Oct. 1965. [Online]. Available: http://doi.acm.org/10.1145/365628.365657

[14] M.-T. Luong, R. Socher, and C. D. Manning, "Better word representations with recursive neural networks for morphology," in *CoNLL*, 2013.

[15] E. Bruni, N.-K. Tran, and M. Baroni, "Multimodal distributional semantics." *J. Artif. Intell. Res.*, vol. 49, no. 1-47, 2014.

[16] K. Radinsky, E. Agichtein, E. Gabrilovich, and S. Markovitch, "A word at a time: Computing word relatedness using temporal semantic analysis," in *Proceedings of the 20th International Conference on World Wide Web.* ACM, 2011, pp. 337–346.

[17] S. Baker, R. Reichart, and A. Korhonen, "An unsupervised model for instance level subcategorization acquisition." in *EMNLP*, 2014, pp. 278–289.

[18] F. Hill, R. Reichart, and A. Korhonen, "Simlex-999: Evaluating semantic models with (genuine) similarity estimation," *Computational Linguistics*, 2015.

[19] D. Yang and D. M. W. Powers, "Verb similarity on the taxonomy of wordnet," in *3rd International WordNet Conference (GWC-06)*, 2006.

[20] A. Almuhareb, "Attributes in lexical acquisition," Ph.D. dissertation, University of Essex, 2006.

[21] M. Baroni and A. Lenci, "How we BLESSed distributional semantic evaluation," in *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics.* Association for Computational Linguistics, 2011, pp. 1–10.

[22] M. Baroni, B. Murphy, E. Barbu, and M. Poesio, "Strudel: A corpus-based semantic model based on properties and types," *Cognitive Science*, vol. 34, no. 2, pp. 222–254, 2010. [Online]. Available: http://dx.doi.org/10.1111/j.1551-6709.2009.01068.x

[23] M. Baroni and A. Lenci, "Distributional memory: A general framework for corpus-based semantics," *Computational Linguistics*, vol. 36, no. 4, pp. 673–721, 2010.

[24] U. Padó, "The integration of syntax and semantic plausibility in a wide-coverage model of human sentence processing," Ph.D. dissertation, Universitätsbibliothek, 2007.

[25] Y. Tsvetkov, M. Faruqui, W. Ling, G. Lample, and C. Dyer, "Evaluation of word vector representations by subspace alignment," in *Proc. of EMNLP*, 2015.

[26] R. Soricut and F. Och, "Unsupervised morphology induction using word embeddings," in *Proc. NAACL*, 2015.

[27] Z. S. Harris, "Distributional structure," *Word*, vol. 10, no. 2-3, pp. 146–162, 1954.

[28] J. A. Bullinaria and J. P. Levy, "Extracting semantic representations from word co-occurrence statistics: A computational study," *Behavior Research Methods*, vol. 39, no. 3, pp. 510–526, 2007.

[29] R. Fabri, M. Gasser, N. Habash, G. Kiraz, and S. Wintner, "Linguistic introduction: The orthography, morphology and syntax of semitic languages," in *Natural Language Processing of Semitic Languages*. Springer, 2014, pp. 3–41.

[30] R. Al-Rfou, B. Perozzi, and S. Skiena, "Polyglot: Distributed word representations for multilingual nlp," in *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*. Sofia, Bulgaria: Association for Computational Linguistics, August 2013. [Online]. Available: http://www.aclweb.org/anthology/W13-3520 pp. 183–192.

[31] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781, 2013.

[32] W. Ling, C. Dyer, A. Black, and I. Trancoso, "Two/too simple adaptations of word2vec for syntax problems," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 1299–1304.

[33] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation.” in *EMNLP*, vol. 14, 2014, pp. 1532–1543.