11-1-2022

# Word Representation: Theoretical Explanation of an Empirical Fact

Leonel Escapita
*The University of Texas at El Paso*, laescapita@miners.utep.edu

Diana Licon
*The University of Texas at El Paso*, dlicon2@miners.utep.edu

Madison Anderson
*The University of Texas at El Paso*, mranderson2@miners.utep.edu

Diego Pedraza
*The University of Texas at El Paso*, dapedraza@miners.utep.edu

Vladik Kreinovich
*The University of Texas at El Paso*, vladik@utep.edu



Comments:

Technical Report: UTEP-CS-22-110

# Word Representation: Theoretical Explanation of an Empirical Fact

Leonel Escapita, Diana Licon, Madison Anderson, Diego Pedraza, and Vladik Kreinovich

**Abstract** There is a reasonably accurate empirical formula that predicts, for two words $i$ and $j$, the number $X_{ij}$ of times when the word $i$ will appear in the vicinity of the word $j$. The parameters of this formula are determined by using the weighted least square approach. Empirically, the predictions are the most accurate if we use the weights proportional to a power of $X_{ij}$. In this paper, we provide a theoretical explanation for this empirical fact.

## 1 Formulation of the Problem

**Background: need to describe the relation between words from natural language by a numerical characteristic.** To better understand and process natural-language texts, computers need to have an accurate precise description of the meaning of different words and different texts. In particular, we need to represent, in a computer, to what extent different words from natural language are related to each other. This relation can be described, e.g., by the number $X_{ij}$ of times when word $i$ appears in the context of word $j$.

**How this characteristic can be estimated: a straightforward approach.** In principle, we can get the values $X_i$ by analyzing several natural-language texts.

The more texts we analyze, the more accurately we can represent this dependence. However, there are so many natural-language texts that it is not feasible to process them all, so we have to limit ourselves to values obtained by processing some of the available texts.

Leonel Escapita, Diana Licon, Madison Anderson, Diego Pedraza, and Vladik Kreinovich
Department of Computer Science, University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA, e-mail: mranderson2@miners.utep.edu,
laescapita@miners.utep.edu, dlicon2@miners.utep.edu, dapedraza@miners.utep.edu,
vladik@utep.edu

**Limitations of the straightforward approach.** In these texts, for words which are closely related – e.g., "cow" and "milk" – we will probably have a sufficient number of situations in which one of these two words appeared in the context of another one. In such situations, the values that we measure based on these texts provide a reasonable statistically accurate description of this relation – and we can use this relation to predict how many pairs we will have if we add additional texts to our analysis.

On the other hand, if the two words are not so closely related, then in the current texts, we may have only a few examples of such pairs. In general, in statistics, when the sample is small, the corresponding estimates are not very accurate and do not lead to good predictions.

**How can we get more accurate estimates: a general idea.** Good news is that, in general, predictions are not only based on statistics – otherwise, we would never be able to predict rare events like solar eclipses – they are also based on the known dependencies between the corresponding quantities. For example, if we want to analyze with what force two charged bodies attract or repel each other, we do not need to perform experiments with all possible pairs of such bodies: we know Coulomb's law according to which we can predict this force if we know the charges of both bodies (and the distance between them). Similarly, Newton's laws allows us to predict the gravitational force between two bodies if we know their masses (and the distance between the bodies).

It is therefore desirable to look for a similar dependence – that would describe the quantity $X_{ij}$ describing the relation between the two words in terms of some numerical characteristics of the two words $i$ and $j$.

**How this general idea is used.** At present, these characteristics are determined by training a neural network; see, e.g., [2] and references therein. There is also a reasonably good approximate analytical formula for describing this dependence (see, e.g., [3]):

$$\ln(X_{ij}) \approx b_i + \widetilde{b}_j + w_i \cdot \widetilde{w}_j,$$

where $b_i$ and $\widetilde{b}_j$ are numbers, $w_i$ and $\widetilde{w}_j$ are vectors, and $a \cdot b$ is dot (scalar) product.

The values $b_i$, $\widetilde{b}_j$, $w_i$ and $\widetilde{w}_j$ can be found by using the Least Squares method, i.e., by solving the minimization problem

$$J \stackrel{\text{def}}{=} \sum_{i,j} f(X_{ij}) \cdot (b_i + \widetilde{b}_j + w_i \cdot \widetilde{w}_j - \ln(X_{ij}))^2 \to \min$$

for an appropriate weight function $f(X)$.

**Empirical fact.** The efficiency of this method depends on the appropriate choice of the weight function $f(X)$. Empirical data shows that the most efficient weight function is the power law $f(X) = X^a$.

**Remaining problem.** How can we explain this empirical fact?

**What we do in this paper.** In this paper, we provide a theoretical explanation for this fact.

## 2 Our Explanation

**Analysis of the problem.** The values $X_{ij}$ depend on the size of the corpus. For example, if we consider twice smaller corpus, each value $X_{ij}$ will decrease approximately by half. In general, if we consider a $\lambda$ times larger corpus, we will get new values which are close to $\lambda \cdot X_{ij}$.

**A natural requirement.** The word representation should depend only on the words themselves, not on corpus size. So, the resulting representation should not change if we replace $X_{ij}$ with $\lambda \cdot X_{ij}$.

**How to describe this requirement in precise terms: discussion.** Of course, if we replace $X_{ij}$ with $\lambda \cdot X_{ij}$, the weights will change. However, this does not necessarily mean that the resulting representations will change.

Namely, for any $c > 0$, optimizing any function $J$ is equivalent to optimizing the function $c \cdot J$. For example, if we are looking for the richest person on Earth, the same person will be selected as the richest whether we count his richness in dollars on in pesos.

So, if we replace $f(X)$ with $c \cdot f(X)$, we will get a new objective function $c \cdot J$ instead of the original objective function $J$, but we will get the same representations $w_i$.

**Resulting requirement.** In view of the above discission, we can have

$$f(\lambda \cdot X) = c \cdot f(X)$$

, and the resulting representation will be the same.

So, the invariance with respect to corpus size can be described as follows: for every real number $\lambda > 0$, there exists a real number $c > 0$ depending on $\lambda$ for which

$$f(\lambda \cdot X) = c(\lambda) \cdot f(X).$$

**What are the consequences of this requirement.** It is known that all measurable solutions to the functional equation $f(\lambda \cdot X) = c(\lambda) \cdot f(X)$ are power laws $f(X) = A \cdot X^a$; see, e.g., [1]. So, the invariant weight function should have the form

$$f(X) = A \cdot X^a.$$

As we have mentioned, multiplying all the values of the objective function by a constant does not change the resulting values. Because of this, the weight function $A \cdot X^a$ and the wight function $X^a$ lead to the same values $w_i$. Thus, it is sufficient to consider the weight function $f(X) = X^a$.

**We get the desired explanation.** This explains why the power law weights work the best: power law weights are the only ones for which the resulting representation does not depend on the corpus size.

## 3 Acknowledgments

This work was supported in part by the National Science Foundation grants:

- 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), and
- HRD-1834620 and HRD-2034030 (CAHSI Includes).

It was also supported by the AT&T Fellowship in Information Technology, and by the program of the development of the Scientific-Educational Mathematical Center of Volga Federal District No. 075-02-2020-1478.

The authors are thankful to the participants of the 2022 UTEP/NMSU Workshop on Mathematics, Computer Science, and Computational Science (El Paso, Texas, November 5, 2022) for valuable discussions.

## References

1. J. Aczél and J. Dhombres, *Functional Equations in Several Variables*, Cambridge University Press, 2008.
2. D. Jurafsky and J. H. Martin, *Speech and Language Processing*, Prentice Hall, Upper Saddle River, New Jersey, 2023.
3. J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation", *Proceedingds of the 2014 Conference on Empirical Methods in Natural Language Processing EMNLP*, Doha, Qatar, October 25–29, 2014, pp. 1532–1543.

## 4 How to Prove the Result About the Functional Equation

The above result about the functional equation s easy to prove when the function $f(X)$ is differentiable. Indeed, suppose that $f(\lambda\cdot X) = c(\lambda)\cdot f(X)$. If we differentiate both sides with respect to $\lambda$, we get

$$X \cdot f'(\lambda \cdot X) = c'(\lambda) \cdot f(X).$$

In particular, for $\lambda = 1$, we get $X \cdot f'(X) = a \cdot f(X)$, where $a \stackrel{\text{def}}{=} c'(1)$, so

$$X \cdot \frac{df}{dX} = a \cdot f.$$

We can separate the variables if we multiply both sides by $\frac{dX}{X \cdot f}$, then we get

$$\frac{df}{f} = a \cdot \frac{dX}{X}.$$

Integrating both sides of this equality, we get

$$\ln(f) = a \cdot \ln(X) + C.$$

By applying $\exp(x)$ to both sides, we get

$$f(X) = \exp(a \cdot \ln(X) + C) = A \cdot X^a, \text{ where } A \stackrel{\text{def}}{=} e^C,$$

which is exactly the desired formula.