tomo suli pi kama sona

NIMI PI TOMO NI

tsbohc

TOKI PONA: DATA-DRIVEN VOCABULARY ANALYSIS

(tenpo ni la, lipu ni li pona ala)

ma pona 2022

CONTENTS

INTRODUCTION

Toki Pona is the second most spoken constructed language in the world. Its core vocabulary consists of only 120-140 words, not including words that are rare and/or considered non-standard by the majority of speakers. Despite the small vocabulary size, Toki Pona can be effectively used to convey a wide range of ideas of varying complexity.

This research aims to perform the semantic analysis and classification of the vocabulary of Toki Pona.

Numerous dictionaries of Toki Pona exist, but only the most immediate word definitions are provided. They may be enough for someone learning the language to grasp the general idea, but are insufficient to reflect the way the words are used in all of their shades of meaning. Because of this, the said analysis cannot be performed without studying how the language is spoken.

One approach to solving this problem would be to develop a system of evaluating the speaker proficiency. The evaluated speakers of the language who passed the test can then be recruited. By the way of discussion every word can then be described in detail.

The primary difficulty of this approach is that the system which would evaluate speakers has to be created by the proficient speakers. In other words, this is a causality dilemma. This approach would also rely on the judgement of a relatively small sample of the community. With this approach, the results would be based on the perception of the usage patterns of the language and not the usage of the language itself.

A different approach is to let the data drive the analysis. An adequately large corpus can be compiled by a single person. Modern approaches to Natural Language Processing can then be used to obtain the data on the language

change and the semantic distribution of the vocabulary. Observations made on this data can then serve as the basis for the semantic analysis of the vocabulary.

- **Subject.** Semantic analysis and classification of vocabulary.
- **Object.** Toki Pona, a constructed language.
- **Goal.** Perform the semantic analysis and classification of the vocabulary of the language.
  - **Problem.** Vocabulary cannot be analysed based on the existing resources as they do not contain sufficient information for analysis.
  - **Solution.** Use natural language processing techniques to construct a semantic model of the language and base the analysis on it.
- **Methodology.** Distributional semantics and natural language processing, namely language modelling (word embedding).

Objectives

1. Define and classify constructed languages.
2. Describe toki pona, its philosophy, history, and unique features.
3. Define distributional semantics.
4. Define modern approaches to Natural Language Processing applicable to the research.
5. Obtain the necessary corpora.
6. Construct a vector space model of the language.
7. Make observations on the model.
8. Classify the vocabulary based on the observed semantic relationships between the words of the vocabulary.

Relevance

With the rise of the internet, constructed languages now have a place where they can live and thrive. Constructed languages are rapidly gaining popularity. Despite this, the only constructed language that has seen much representation in scientific writing is Esperanto.

The existing dictionaries of Toki Pona can greatly benefit from the findings of this research. This data can also be used as an aid in teaching the language to new speakers.

The Vector Space Model of Toki Pona constructed in the course of this research can find further use in information retrieval, topic modelling, text prediction, sentiment analysis, and many other areas.

DISTRIBUTIONAL SEMANTICS AND CONSTRUCTED LANGUAGES

1 Natural language processing

"Linguistics is concerned not only with language per se, but must also deal with how humans model the world. The study of semantics, for example, must relate language expressions to their meanings, which reside in the mental models possessed by humans. <...> Whereas computational linguistics, as a subfield of linguistics, is concerned with the formal or computational description of rules that languages follow" [Tsujii 2021].

The aim of this research is to bridge the gap between the two disciplines, to use computational linguistics to build a semantic model of a constructed language. This model can then be used to explore the nuances of how humans speak the said language.

In turn, "Natural Language Processing is a field at the intersection of computer science, artificial intelligence, and linguistics" [Vajjala, Majumder 2020, p. 7]. "Natural language processing includes a range of algorithms, tasks, and problems that take human-produced text as an input and produce some useful information, such as labels, semantic representations, and so on, as an output" [Hagiwara 2021, p. 4].

2 Distributional semantics

The core idea behind distributional semantics has roots in American structuralism (Harris) and British lexicology (Firth) and is known as the distributional hypothesis. In its simplest form, it states that "similarity in

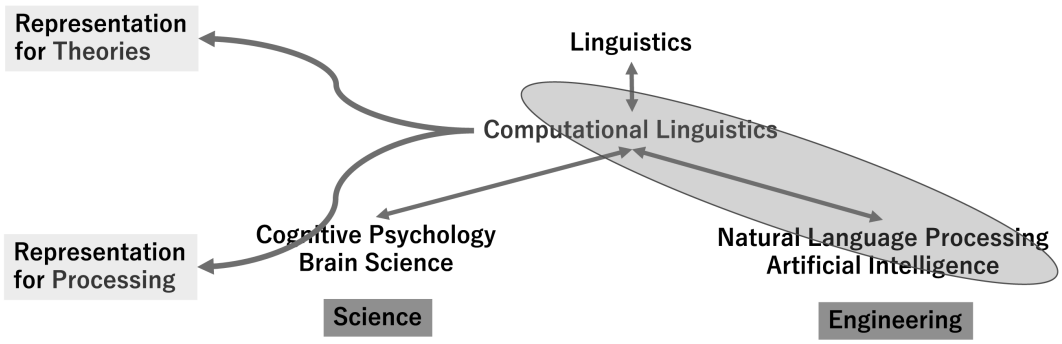

Figure .1: Language-related disciplines [Tsujii 2021]

meaning results in similarity of linguistic distribution" [Harris 1954].

The reverse of this statement is also true. That is, meaning can be inferred from context. The aim of distributional semantics is exactly that, to learn the meanings of linguistics units from a corpus of text. That is, distributional semantics can be described as the field that reverse-engineers the distributional hypothesis.

Distributional semantics was popularised by Firth in the 1950s. In a 1957 publication he wrote, "the placing of a text as a constituent in a context of situation contributes to the statement of meaning since situations are set up to recognise use. <...> You shall know a word by the company it keeps!" [Firth 1957, p. 11].

The ideas introduced by the distributional hypothesis have received attention in cognitive science [Mcdonald 2008] and language learning [Yarlett, Ramscar, Dye 2008].

Overview

Distributional semantics has become widespread with the adoption of information technology in the field of linguistic research.

Distributional semantics are most frequently applied by taking large amounts of text as input and pushing it through an abstraction algorithm to produce a distributional model as output.

Planets of the solar system are orbiting the *sun*. The *moon* is orbiting the earth. It's his antique *typewriter* clacking. <...>

⟶ algorithm

	dim1	dim2
sun	0.11023	0.53848
moon	0.21575	0.44034
typewriter	0.52834	0.05389

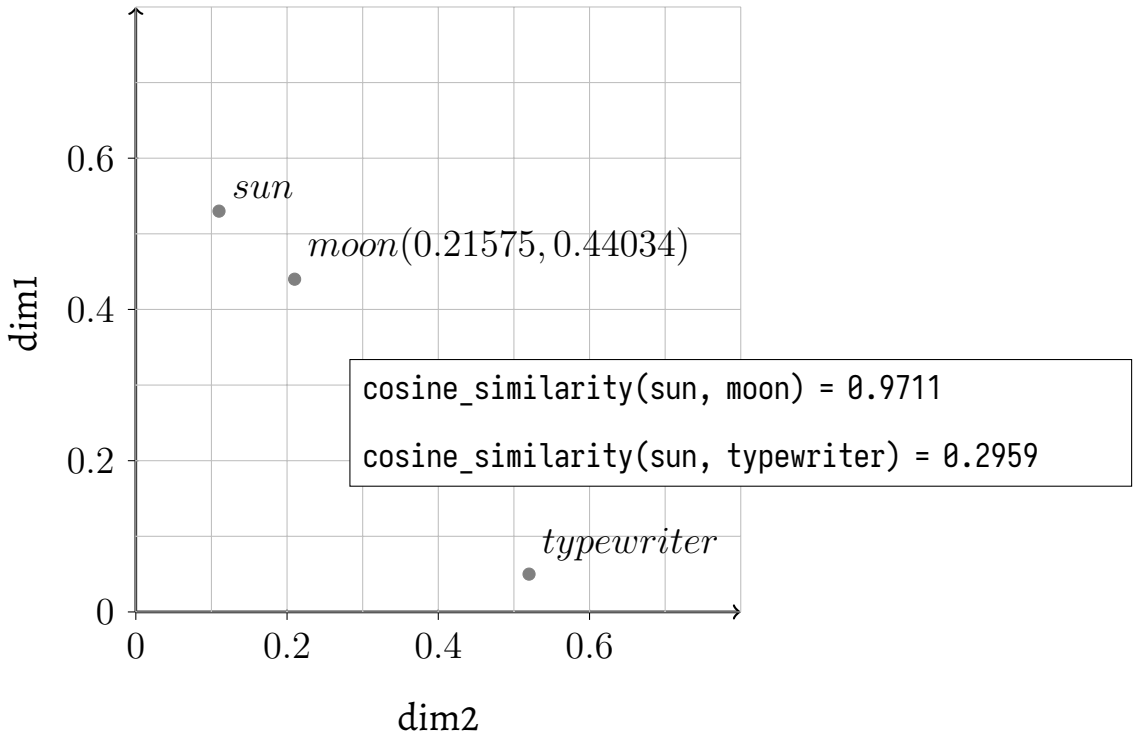

Figure .2: Distributional semantics, an illustrated overview

In the model, the semantic representations are stored in the form of vectors. Vectors are essentially lists of numbers that refer to points in a multi-dimensional space. These vectors are referred to as word vectors.

The position of a word in the semantic space is determined by the values in its word vectors, that is, the context the word is most commonly used in. This way, words that are often used together are placed close to each other. The distance between points is a measurement of semantic similarity between the words they represent.

The multi-dimensionality of the word vector encodings can be reduced to

only two or three dimensions. The resulting dimensions can then be used to create a projection of the model which can be observed by the human eye.

All of the approaches to distributional semantics share the quality of learning semantic representations from a corpus in an unsupervised manner. With a sufficiently large corpus, this excludes any but the collective human modelling of the world from the data.

2.1 Word vectors

"Word vectors represent words as multidimensional continuous floating point numbers where semantically similar words are mapped to proximate points in geometric space" [Ahire 2018].

In simpler terms, a word vector is a numerical representation of a word in a corpus relative to every other word in that corpus.

"Word vectors are numerical vector representations of word semantics, or meaning, including literal and implied meaning, meaning that word vectors can capture the connotation of words. And they combine all that into a dense vector (no zeros) of floating point values. This dense vector enables queries and logical reasoning" [Lane, Hapke, Howard 2019, p. 182].

Essentially, word vectors are the means by which the aforementioned semantic distribution of words can be represented numerically. That is, a representation that is easy for a computer to understand and operate on.

2.1.1 Measuring semantic similarity

The semantic similary between two vectors is primary measured in two ways: using cosine similarity or euclidian distance.

The primary advantage of using one of these two methods is that they can be calculated for vectors of any dimensionality.

Euclidean distance

The Euclidean distance between two points is the length of a line segment between the two points. It can also be defined as the shortest distance between two points in an n-dimensional space. For the purposes of calculating the Euclidean distance, the vectors are viewed as point coordinates.

$$d_{Euc}(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + ... + (p_n - q_n)^2}$$

Cosine similarity

Cosine similarity is a measurement of similarity between two sequences of numbers. When calculating cosine similarity, the two sequences of numbers are viewed as vectors. Cosine similarity is equal the cosine of the angle between two vectors, that is, the dot product of the vectors devided by the product of their lengths.

Cosine similarity always falls into the interval $[-1, 1]$. Two parallel vectors have a cosine similarity of 1, two orthogonal (perpendicular to each other) vectors have a cosine similarity of 0, while two opposite vectors have a cosine similarity of -1.

$$s_{cos}(A,B) := cos(\theta) = \frac{A \cdot B}{||A|| \cdot ||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

This method was chosen to simplify the process of comparing similarities between vector pairs. Where the Euclidean distance provides an absolute value, the cosine similarity provids a fraction.

2.2 Implementations

One of the earliest implementations featured a simple counting algorithm. For each word in a corpus, the frequencies of each of the context words were counted. The counts were then normalised by the total number of collected word relationships [Erk 2012].

2.3 Word vectors

"Word vectors represent words as multidimensional continuous floating point numbers where semantically similar words are mapped to proximate points in geometric space" [Ahire 2018].

In simpler terms, a word vector is a numerical representation of a word in a corpus relative to every other word in that corpus.

"Word vectors are numerical vector representations of word semantics, or meaning, including literal and implied meaning. So word vectors can capture the connotation of words, like *peopleness, animalness, placeness, thingness,* and even *conceptness*. And they combine all that into a dense vector (no zeros) of floating point values. This dense vector enables queries and logical reasoning" [Lane, Hapke, Howard 2019, p. 182].

Word embedding

3 Constructed languages

3.1 The notion of a constructed language

3.2 Classification

LANGUAGE MODELLING AND TOKI PONA

1 Toki Pona

1.1 History

1.2 Phonology

1.3 Grammar

1.4 Vocabulary

2 Vector space model

2.1 Corpus acquisition

2.2 Text normalisation

2.3 Model construction

2.4 Projection and visualisation

2.5 Observations

REFERENCES

1. *Ahire J. B.* Introduction to Word Vectors // Retrieved March. — 2018. — Vol. 12. — P. 2018.
2. *Erk K.* Vector Space Models of Word Meaning and Phrase Meaning: A Survey // Language and Linguistics Compass. — 2012. — Oct. — Vol. 6. — DOI: 10.1002/lnco.362.
3. *Firth J.* A Synopsis of Linguistic Theory, 1930-1955. — 1957. — URL: https://books.google.ru/books?id=T8LDtgAACAAJ.
4. *Hagiwara M.* Real-World Natural Language Processing: Practical Applications with Deep Learning. — Manning, 2021. — ISBN 9781617296420. — URL: https://books.google.ru/books?id=0k5NEAAAQBAJ.
5. *Harris Z. S.* Distributional Structure // WORD. — 1954. — Vol. 10, no. 2/3. — P. 146–162. — DOI: 10.1080/00437956.1954.11659520. — eprint: https://doi.org/10.1080/00437956.1954.11659520. — URL: https://doi.org/10.1080/00437956.1954.11659520.
6. *Lane H., Hapke H., Howard C.* Natural Language Processing in Action: Understanding, analyzing, and generating text with Python. — Manning Publications, 2019. — ISBN 9781617294631. — URL: https://books.google.ru/books?id=UyHgswEACAAJ.
7. *Mcdonald S.* Testing the distributional hypothesis: The influence of context on judgements of semantic similarity. — 2008. — July.
8. *Tsujii J.* Natural Language Processing and Computational Linguistics // Computational Linguistics. — 2021. — Dec. — Vol. 47, no. 4. — P. 707–727. — ISSN 0891-2017. — DOI: 10.1162/coli_a_00420. — eprint: https:

//direct.mit.edu/coli/article-pdf/47/4/707/1979478/coli_a_00420.pdf. — URL: https://doi.org/10.1162/coli%5C_a%5C_00420.

9. *Vajjala S., Majumder B.* Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems. — O'Reilly Media, 2020. — ISBN 9781492054009. — URL: https://books.google.ru/books?id=hPrrDwAAQBAJ.

10. *Yarlett D., Ramscar M., Dye M.* Language Learning Through Similarity-Based Generalization. — 2008. — Jan.