UNIVERSITY OF TURKU

Peepoo Robot Language

BSc Omar Mayani

MSc thesis
June 2026

MATEMATIIKAN JA TILASTOTIETEEN LAITOS

Reviewers:
Prof. H. H.
PhD D.D.

UNIVERSITY OF TURKU, Department of Mathematics and Statistics

MSc Thesis
Subject: Mathematics
Author: Omar Mayani
Title: Peepoo Robot Language
Supervisor: Prof. H. H.
Pages: xx pages + xx appendix pages
Month and year: June 2026

Write the abstract here. Insert the \noindent instruction in the beginning of the paragraph to prevent the first line indention in LaTeX.

Command \vspace leaves space between the paragraphs, 4mm looks good.

Write the abstract compactly, avoiding unnecessary repetition.

Keywords: Abstract, MSc thesis, LaTeX system.

Contents

1 Introduction

This section includes general despripition of the field and the topic of the MSc thesis.

Note that if you have used AI in your writing, you must mention it. Check for guidelines https://utuguides.fi/artificialintelligence. Good place to mention that AI tolls have been used is here, for example, in the following form:

The ChatGPT AI has been used to enhance the language of the thesis.

2 Foundations of Statistical Learning for NLP

Theorem 1. *Assume that in interval $[a, b]$ $F'(x) = f(x)$ holds and that f is continuous.*[1] *Then*

$$\int_a^b f(x)\,dx = F(b) - F(a).$$

[1] The continuity is essential here.

3 Representation Learning

3.1 Introduction

Machines need to understand each word first so as to master the sophisticated meanings of human languages. Hence, effective word representations are essential for natural language processing (NLP), and it is also a good start for introducing representation learning in NLP.

The form of a word representation can be divided into two categories: symbolic and distributed representations. In symbolic representation, each word is represented as a unique symbol or index, such as one-hot encoding. In distributed representation, each word is represented as a dense vector in a continuous vector space, where semantically similar words are mapped to nearby points.

Symbolic representations are easy to implement and interpret, but they suffer from the curse of dimensionality and cannot capture semantic relationships between words. The distributed word representation overcomes these problems by representing words as low-dimensional real-valued dense vectors.

In distributed word representation, each dimension in isolation is meaningless because semantics is distributed over all dimensions of the vector. Distributed representation can be obtained by factorizing the matrices of symbolic representations, such as in Latent Semantic Analysis LSA, or by optimizing the word vectors with gradient descent to predict the context words, such as in Word2Vec. Glove is a hybrid method that factorizes the co-occurrence matrix of words while also optimizing the word vectors to predict the co-occurrence counts.

One-hot

$$\mathbf{w_j^{(i)}} = \begin{cases} 1, & \text{if } j = i, \\ 0, & \text{otherwise,} \end{cases}$$

3.2 Distributed Word Representations

Although simple and interpretable, symbolic representations are not the best choice for computation. For example, the very sparse nature of the symbolic representation makes it difficult to compute word-to-word similarities.

The difficulty of symbolic representation is solved by the distributed representation. Distributed representation represents a subject (a word in our case) as a fixed-length real-valued vector, where no clear meaning is assigned to any single dimension of the vector. More specifically, semantics is scattered over all (or a large portion) of the dimensions of the representation, and one dimension contributes to the semantics of all (or a large proportion) of the words

3.2.1 Matrix Factorization-based Approaches

Latent Semantic Analysis (LSA) lsalsa

Probabilistic LSA plsaplsa

Latent Dirichlet Allocation (LDA) IF APPLIED LATER

3.2.2 Word2Vec and GloVe

Word2Vec Word2Vec adopts the distributional hypothesis but does not take a count-based approach. It directly uses gradient descent to optimize the representations of a word toward tis neighbors' representations. There are two main architectures for Word2Vec: Continuous Bag of Words (CBOW) and Skip-Gram. The difference is that CBOW predicts the target word based on multiple context words, while Skip-Gram predicts the context words based on the center word.

Formally, CBOW predicts the word w_i as

$$P(w_i|w_{i-l}, \dots, w_{i-1}, w_{i+1}, \dots, w_{i+l}) = \sigma \left(\mathbf{W} \sum_{\substack{j=i-l \\ j\neq i}}^{i+l} \mathbf{w}_j \right),$$

where $2l+1$ is the context size, σ is the softmax function, $\mathbf{W} \in \mathbb{R}^{|V|\times m}$ is the weight matrix and $\mathbf{w}_j$ is the embedding of word w_j.

The CBOW model is optimized by minimizing the sum of the negative log probabilities:

$$\mathcal{L} = -\sum_{i=1}^{N} \log P(w_i|w_{i-l}, \dots, w_{i-1}, w_{i+1}, \dots, w_{i+l}).$$

Contrary to CBOW, Skip-Gram predicts the context words based on the center word. Formally, given a word w_i, Skip-Gram predicts the context words as

$$P(w_j|w_i) = \sigma \left(\mathbf{W}\mathbf{w}_i^\intercal \right), \quad j \in \{i-l, \dots, i-1, i+1, \dots, i+l\}.$$

GloVe

References

[1] Newton and Leibniz

[2] Riemann