Department of Information and Computer Science

Computational Modeling and Simulation of Language and Meaning: Similarity-Based Approaches

Tiina Lindh-Knuutila

Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 49/2014

Computational Modeling and Simulation of Language and Meaning: Similarity-Based Approaches

Tiina Lindh-Knuutila

A doctoral dissertation completed for the degree of Doctor of Science in Technology to be defended, with the permission of the Aalto University School of Science, at a public examination held at the lecture hall T2 of the school on 9 May 2014 at 12.

Aalto University
School of Science
Department of Information and Computer Science

Supervising professor
Aalto Distinguished Professor Erkki Oja

Thesis advisors
Professor Timo Honkela
Dr. Mathias Creutz

Preliminary examiners
Docent Hanna Suominen, NICTA, Australia
Dr. John A. Bullinaria, University of Birmingham, UK

Opponents
Professor (Emeritus) Fred Karlsson, University of Helsinki, Finland
Professor Ari Visa, Tampere University of Technology, Finland

Aalto University publication series
DOCTORAL DISSERTATIONS 49/2014

ISBN 978-952-60-5643-2
ISBN 978-952-60-5644-9 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-5644-9

Images: The cover image is a derivative of
http://www.wikihow.com/Image:Draw-a-Book-Step-12.jpg by
Wikiphoto, used under CC BY-NC-SA 3.0

Unigrafia Oy
Helsinki 2014

Finland

441 697
Printed matter

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Tiina Lindh-Knuutila

Name of the doctoral dissertation
Computational Modeling and Simulation of Language and Meaning: Similarity-Based Approaches

Publisher School of Science

Unit Information and Computer Science

Series Aalto University publication series DOCTORAL DISSERTATIONS 49/2014

Field of research Speech and Language Technology

Manuscript submitted 20 January 2014 **Date of the defence** 9 May 2014

Permission to publish granted (date) 2 April 2014 **Language** English

☐ **Monograph** ☒ **Article dissertation (summary + original articles)**

Abstract

This dissertation covers various similarity-based, data-driven approaches to model language and lexical semantics. The availability of large amounts of text data in electronic form allows the use of unsupervised, data-driven methodologies. Compared to linguistic models based on expert knowledge, which are often costly or unavailable, the data-driven analysis is faster and more flexible. The same methodologies can be often used regardless of the language. In addition, data-driven analysis may be exploratory and offer a new view on the data.

The complexity of different European languages was analyzed at syntactic and morphological level using unsupervised methods based on compression and unsupervised morphology induction. The results showed that the unsupervised methods are able to produce useful analyses that correspond to linguistic models.

The distributional word vector space models represent the meaning of words in a text context of co-occurring words, collected from a large corpus. The vector space models were evaluated with linguistic models and human semantic similarity judgment data. Two unsupervised methods, Independent Component Analysis and Latent Dirichlet Allocation, were able to find groups of semantically similar words, corresponding reasonably well to the evaluation sets. In addition to validating the results of the unsupervised methods with the evaluation data, the research was also exploratory. The unsupervised methods found semantic word sets not covered by the evaluation set, and the analysis of the categories of the evaluation sets showed quality differences between the categories.

In the agent simulation models, the meaning of words was directly linked to the perceived context of the agent. Each agent had a subjective conceptual memory, in which the associations between words and perceptions were formed. In a population of simulated agents, the emergence of a shared vocabulary was studied through simulated language games. As a result of the simulations, a shared vocabulary emerges in the community.

Keywords lexical semantics, language, meaning, computational modeling, vector space models, language complexity, agent simulation, unsupervised learning, machine learning

ISBN (printed) 978-952-60-5643-2 **ISBN (pdf)** 978-952-60-5644-9

ISSN-L 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2014

Pages 310 **urn** http://urn.fi/URN:ISBN:978-952-60-5644-9

Tekijä
Tiina Lindh-Knuutila

Väitöskirjan nimi
Kielen ja merkityksen laskennallinen mallintaminen ja simulointi: samankaltaisuuteen perustuvia menetelmiä

Julkaisija Perustieteiden korkeakoulu

Yksikkö Tietojenkäsittelytieteen laitos

Sarja Aalto University publication series DOCTORAL DISSERTATIONS 49/2014

Tutkimusala Puhe- ja kieliteknologia

Käsikirjoituksen pvm 20.01.2014 **Väitöspäivä** 09.05.2014

Julkaisuluvan myöntämispäivä 02.04.2014 **Kieli** Englanti

☐ **Monografia** ☒ **Yhdistelmäväitöskirja (yhteenveto-osa + erillisartikkelit)**

Tiivistelmä

Tämä väitöskirja kattaa useita samankaltaisuuteen perustuvia datalähtöisiä menetelmiä, joita käytetään kielen ja merkityksen mallintamiseen. Suuret, sähköisessä muodossa olevat tekstiaineistot mahdollistavat ohjaamattomien datalähtöisten menetelmien käytön. Verrattuna asiantuntijoiden tuottamiin lingvistisiin malleihin, jotka ovat usein kalliita tai joita ei aina ole saatavilla, datalähtöinen analyysi on nopeampaa ja usein joustavampaa. Samat menetelmät sopivat usein kielestä riippumatta. Lisäksi datalähtöinen analyysi voi olla eksploratiivista ja siten tarjota uuden näkökulman aineistoon.

Tässä työssä analysoitiin useiden eurooppalaisten kielten syntaktisen ja morfologisen tason kompleksisuutta ohjaamattomilla menetelmillä, jotka perustuvat datan kompressioon ja ohjaamattomaan morfologian oppimiseen. Tulokset osoittavat, että ohjaamattomat menetelmät tuottavat hyödyllisiä tuloksia, jotka vastaavat lingvistisiä malleja.

Jakaumiin perustuvat sana-avaruusmallit (Vector Space Models) käyttävät sanojen merkityksen esittämiseen sanojen kontekstia eli sanojen välisiä yhteisesiintymiä, jotka kerätään laajoista tekstiaineistoista. Tässä työssä käytettiin sana-avaruusmalleja, joita evaluoitiin käyttäen lingvistisiä malleja ja semanttisia evaluaatioaineistoja. Työssä käytettiin kahta ohjaamatonta menetelmää, riippumattomien komponenttien analyysia (Independent Component Analysis) sekä latenttia Dirichlet-allokaatiota (Latent Dirichlet Allocation), joilla löydettin semanttisesti samankaltaisia sanajoukkoja, jotka vastasivat kohtuullisen hyvin evaluaatioaineistoja. Evaluaatiotulosten lisäksi tutkimuksessa oli myös eksploratiivinen komponentti. Ohjaamattomat menetelmät löysivät merkitykseltään samankaltaisia sanajoukkoja, jotka puuttuivat evaluaatioaineistoista. Lisäksi menetelmillä löydettiin laadullisia eroja kategorioiden välillä.

Agenttisimulaatiomallissa sanojen merkitys liittyi suoraan agentin havaitsemaan kontekstiin. Jokaisella agentilla oli oma subjektiivinen käsitemuisti, jossa assosiaatiot sanojen ja havaintojen välillä muodostuivat. Tässä työssä jaetun kielen syntyä tutkittiin useiden simuloitujen agenttien muodostamassa populaatiossa, jossa agentit kommunikoivat simuloituja kielipelejä käyttäen. Simulaatiokokeiden tuloksena jaettu kieli syntyy agenttipopulaatiossa.

Avainsanat leksikaalinen semantiikka, kieli, merkitys, laskennallinen mallintaminen, vektoriavaruusmalli, kielen kompleksisuus, agenttisimulaatio, ohjaamaton oppiminen, koneoppiminen

ISBN (painettu) 978-952-60-5643-2 **ISBN (pdf)** 978-952-60-5644-9

ISSN-L 1799-4934 **ISSN (painettu)** 1799-4934 **ISSN (pdf)** 1799-4942

Julkaisupaikka Helsinki **Painopaikka** Helsinki **Vuosi** 2014

Sivumäärä 310 **urn** http://urn.fi/URN:ISBN:978-952-60-5644-9

Preface

This work has been carried out at Aalto University, Department of Information and Computer Science, and its previous incarnation, the Laboratory of Computer and Information Science. During this time, I have benefited from the supportive and encouraging atmosphere of the department and its personnel. I have also been an adjunct member of two graduate schools: The Language Technology Graduate School (KIT), and the Helsinki Graduate School of Computer Science and Engineering (HeCSE), which have offered me mentoring, courses and a network of people with similar research interests.

This work has been mostly funded by the Department of Information and Computer Science, and the Adaptive Informatics Research Centre (AIRC), located at the department. In addition, I have been funded by the EU project MedIEQ (January 2006–July 2006), two Tekes projects *Kulta 1* (April 2007– December 2008) and *Kulta 2* (January 2009–September 2010), which also allowed me to make a six month research visit to International Computer Science Institute at the University of California at Berkeley. I am also very grateful for the Finnish Cultural Foundation for funding of one year (September 2012–August 2013) for the finalization of this doctoral dissertation. During my doctoral studies, I have also received smaller personal grants from Emil Aaltonen Foundation and the Tekniikan edistämissäätiö (TES) and travel grants from the Helsinki Graduate School of Computer Science and Engineering (HeCSE).

I am deeply indebted to my supervisor, prof. Erkki Oja, who has given me support, both financial and academic, patiently offering his insight to actually get this book into its final form. My instructors, Professor Timo Honkela and Dr. Mathias Creutz, have also given me their time and expertise. I have greatly benefited from collaboration with Timo during all these years, starting from my Master's thesis work, whereas Mathias

has given me his support in the various parts of writing the thesis. While not an official instructor of this dissertation, Dr. Krista Lagus has also offered me her insight at different stages of this work. Thank you all.

All of my work has been very much team work. I am thankful to my co-authors, Professor Timo Honkela, Dr. Krista Lagus, Jaakko Väyrynen, Dr. Mari-Sanna Paukkeri, Dr. Ville Könönen, Juha Raitio, Markus Sadeniemi and Dr. Kimmo Kettunen. This dissertation would not exist without you all.

The former and current doctoral students of the Computational Cognitive Systems research group have been my companions through the hard and the fun parts of the academic life. I especially want to mention Jaakko Väyrynen, Dr. Mari-Sanna Paukkeri, Oskar Kohonen, Dr. Sami Virpioja, Mikaela Kumlander, Matti Pöllä, Marcus Dobrinkat, and Paul Wagner. I have learned a lot from you. It has been a pleasure to work with you.

Several people have lent me their expertise and given comments on the preliminary versions of this manuscript. Thank you, Timo, Krista, Erkki, Mathias, Sami and Oskar for making my work considerably better. I would also like to thank my preliminary examiners Docent Hanna Suominen and Dr. John A. Bullinaria for their valuable comments, which have helped me to clarify the points made in this thesis even more.

In addition to supportive work environment, I have had wonderful support at the home front. Marko, you have kept the home life running in the eve of deadlines, when I disappeared to work for evenings and weekends. Maria and Julia have given me a reminder of the importance of life outside Academia. Observing them growing up and learning how to deal with the world has also given me some useful insight for my research work. From my parents, Tuulikki and Jouko, I have inherited the curiosity that drives my research. This work is dedicated to you.

Espoo, April 8, 2014,

Tiina Lindh-Knuutila

Contents

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I Markus Sadeniemi, Kimmo Kettunen, Tiina Lindh-Knuutila and Timo Honkela. Complexity of European Union languages: A comparative approach. *Journal of Quantitative Linguistics*, Vol. 15, No. 2, pages 185–211, April 2008.

II Jaakko J. Väyrynen and Tiina Lindh-Knuutila. Emergence of multilingual representations by independent component analysis. In *The Ninth Scandinavian Conference on Artificial Intelligence (SCAI 2006)*, Espoo, Finland, pages 101–105, Finnish Artificial Intelligence Society, October 2006.

III Timo Honkela, Tiina Lindh-Knuutila and Krista Lagus. Measuring Adjective Spaces. In *The International Conference on Artificial Neural Networks (ICANN 2010), LNCS Vol. 6352*, Athens, Greece, pages 351–355, September 2010.

IV Tiina Lindh-Knuutila, Jaakko J. Väyrynen and Timo Honkela. Semantic analysis in word vector spaces with ICA and feature selection. In *The 11th Conference on Natural Language Processing (KONVENS)*, Vienna, Austria, pages 98–107. ÖGAI, September 2012.

V Tiina Lindh-Knuutila and Timo Honkela. Exploratory text analysis: Data-driven versus human semantic similarity judgments. In *The International Conference on Adaptive and Natural Computing Algorithms (ICANNGA'13), LNCS Vol. 7824*, Lausanne, Switzerland, pages 428–437, April 2013.

VI Tiina Lindh-Knuutila and Timo Honkela. Exploratory analysis of semantic categories: Comparing data-driven and human similarity judgments. *Submitted to Computational Cognitive Science*, 26 pages, January 2014.

VII Tiina Lindh-Knuutila, Timo Honkela and Krista Lagus. Simulating meaning negotiation using observational language games. In *The Workshop on the Emergence and Evolution of Linguistic Communication (EELC 2006), LNCS Vol. 4211*, Rome, Italy, pages 168–179, September 2006.

VIII Tiina Lindh-Knuutila, Juha Raitio and Timo Honkela. Combining self-organizing and Bayesian models of concept formation. In *Proceedings of the Eleventh Neural Computation and Psychology Workshop, Progress in Neural Processing, Vol. 18, Connectionist Models of Behaviour and Cognition II*, Oxford, UK, pages 193–204, July 2009.

IX Timo Honkela, Ville Könönen, Tiina Lindh-Knuutila and Mari-Sanna Paukkeri. Simulating processes of concept formation and communication. *Journal of Economic Methodology*, Vol. 15, No. 3, pages 245–259, September 2008.

List of Figures

List of Tables

List of Acronyms

AI	Artificial Intelligence.
BLESS	Baroni-Lenci Evaluation of Semantic Spaces.
BMU	Best Matching Unit.
CD	Cardinal number.
EU	European Union.
GTM	Generative Topographic Map.
ICA	Independent Component Analysis.
ILM	Iterated Learning Model.
IR	Information Retrieval.
JJ	Adjective in base form.
LDA	Latent Dirichlet Allocation.
LSA	Latent Semantic Analysis.
LSI	Latent Semantic Indexing.
MAP	Maximum a Posteriori.
MDL	Minimum Description Length.
MDS	Multi-Dimensional Scaling.
NCD	Normalised Compression Distance.
NeRV	Neighbor Retrieval Visualizer.
NLP	Natural Language Processing.
NN	Singular or mass noun.
NNP	Singular proper noun.
NNS	Plural noun.
O	Object.

PCA Principal Component Analysis.
PDP Parallel Distributed Processing.
pLSI Probabilistic Latent Semantic Indexing.
PMI Pointwise Mutual Information.
POS Part of Speech.
PPMI Positive Pointwise Mutual Information.
Pr Precision.

RB Adverbial in base form.
Re Recall.
RGB Red-Green-Blue.

S Subject.
SOM Self-Organizing Map.
SVD Singular Value Decomposition.

tf.idf Term Frequency - Inverse Document Frequency.

V Verb.
VB Verb in base form.
VBG Verb in -ing form.
VBN Verb in past participle.
VSM Vector Space Model.

List of symbols

X, Y	Random variables
A, B	Scalars
a,b,α, β	Scalars
$\mathbf{a}$,$\mathbf{b}$	Vectors
$\mathbf{A}$,$\mathbf{B}$	Matrices
$\mathbf{a}^T$	Transpose
$\|\mathbf{a}\|$	The L_1 norm of $\mathbf{a}$
$\|\|\mathbf{a}\|\|$	The L_2 or Euclidean norm of $\mathbf{a}$
$p(A)$	Probability of an event A
$p(x)$	Probability distribution for X
μ	Mean
σ^2	Variance
$E\{f(x)\}$	Expectation of $f(x)$
$\mathcal{N}(x)$	Gaussian distribution
$H(X)$	Entropy of X
$I(X;Y)$	Mutual information between X and Y
$J(Y)$	Negentropy of Y
w_i	Word or term in a document or vocabulary
d_j	Document in a collection
m	Morpheme in a collection
Σ	Alphabet
N_d	Number of documents in a collection
N_w	Number of words in a vocabulary
$f_j(w_i)$	Term frequency; frequency of a term w_i in document d_j
$df(w_i)$	Document frequency; the number of documents term w_i appears in

$cf(w_i)$	Collection frequency; the number of occurrences of term w_i in the whole collection
T	Model size, number of topics or (independent) components
$\mathcal{M}$	Model
l_i	Language i
$C(l_i)$	Complexity score for language l_i
$V(l_i)$	Size of a file in bytes in language l_i
ρ	Spearman's rank correlation coefficient
p	Level of significance, p-level
$\mathcal{C}, \mathcal{D}, \mathcal{S}$	Spaces
d_ω, d_λ	Distance measures
σ	Association weight
$\boldsymbol{\rho}$	The set of prototypical referents
N_ρ	Number of referents
a_i	The i-th agent
N_a	Number of agents
$\boldsymbol{m}_i$	The i-th map unit on the Self-Organizing Map
R_i	The neighborhood of the map unit m_i
$\|R\|$	The size of the neighborhood
dog	The concept of dog
'dog'	The word form for the concept *dog*
ANIMAL	The category in which concepts such as *dog* belong

1. Introduction

This dissertation belongs to the interdisciplinary fields of computational and cognitive linguistics. The main topic of it is language: an amazing and complex system we humans learn to use effortlessly. In this work, language is explored from different angles, using insight from interrelated research fields, from Linguistics, Semiotics and Cognitive Science to Information and Computer Science, sometimes bordering Philosophy.

The most important aspect of language is meaning, the underlying message conveyed with words, as "meaning is what language is all about" (Langacker, 1987, p. 12). In Linguistics, the field dedicated to studying meaning is called Semantics, and the majority of this dissertation is dedicated to this topic: how to model and represent meaning? Here, the focus is on computational methodologies, which allow us to build models and gain understanding of languages using large data sets and advanced machine learning methods. The purpose of this work is to give an introduction to the problem of computational modeling of language and meaning, along with a set of theoretical and empirical alternatives to approach this problem including considerations on their applicability.

1.1 Computational modeling of language and meaning

A way to try to understand language is to model it. Modeling, let alone computational modeling of meaning is a difficult task. Being humans, we use words to convey meanings effortlessly, and children learn to use them at an early age. At the same time, trying to describe what meaning exactly is, or build a model to describe it is difficult. Still, our computational models for linguistic behavior would be incomplete without an account of meaning (Sahlgren, 2006).

The core methodology chosen in this thesis for computational modeling

of language and meaning is (statistical) machine learning from large text corpora. Machine learning refers to techniques that enable computers to learn to solve specific tasks. With the powerful computing systems that start to be a norm, statistical machine learning methodologies allow us to process and analyze large data sets and find relevant structure from data, including textual data.

The amount of text available in electronic form has grown exponentially over the last years. There are billions of web pages in the World Wide Web, and at the time of writing, over four million articles in Wikipedia, the free Encyclopedia[1]. These type of resources offer us a great sample of natural language to analyze and create models from. In this dissertation, the majority of the methods fall into the category of *unsupervised* learning: methods that find regularities from a set of observations without a predefined set of labels or classes to classify the observations into.

The computational models of meaning used in this thesis employ the concept of the *context*. Andrews et al. (2009) define two major types of statistical data from which semantic representations can be learned: distributional and experiential. Distributional data describes the distribution of words across spoken and written language, whereas the experiential data refers to data that is derived through experience of the physical world. The distributional models represent the similarity of meaning of words using the context of co-occurring words in spoken or written language. The vector space models used in this thesis are an example of a method based on distributional data. The context can also be the sensory information obtained from the environment. This approach is used to represent meaning in the multi-agent simulation experiments, in which the context is the perceived environment of the simulated agent.

This dissertation employs the notion of *similarity* in a (metric) space, at different levels of language. This approach can be contrasted to analyzing similarity in structured representations, such as graphs. The similarity measures are used in comparing the complexity of different natural languages; in measuring the similarity in meaning using distributional vector space models; and when using a geometrical model to represent meaning in the conceptual memory of a simulated agent.

[1] `http://en.wikipedia.org/wiki/Main_page`, accessed December 5, 2013

1.2 The scope and contributions of the thesis

This thesis addresses a multidisciplinary topic of modeling language and meaning, using largely unsupervised machine learning methods, concentrating on written language. The contributions of this work are in three related areas. At the broadest level, this dissertation analyzes language as a system, comparing different natural languages. We pose the following question: Can unsupervised methods be used in analyzing the differences and similarities of complexity of different languages?

The majority of this dissertation is dedicated to the use of natural language data as a resource to build distributional, corpus-based semantic representations. At this point, we narrow the analysis down into one natural language or a pair of languages. This work will concentrate on *lexical meaning*, that is, the meaning of individual words. In this part, answers to several research questions are searched for. How to build and evaluate different corpus-based semantic representations? Can we find semantically similar word groups or categories that match human semantic similarity judgments? What can be said of the quality of the categories of the evaluation sets?

In the final part, the analysis narrows down even further, to the level of individuals and small communities, and the object of study is language emergence and meaning negotiation. More specifically, we ask the question: can a shared vocabulary develop in a population of learners, each with their own subjective semantic memory? For this purpose, an agent simulation model is developed, and the meaning of individual words is represented through simple observations from environment. In this part of the thesis, the focus is not on an existing human language, but instead, on the adaptive processes that are used when the agents create a vocabulary of their own, shared language. In the following, each of these domains are discussed in more detail.

1.2.1 Language as a system: Complexity at the level of languages

In this dissertation, the complexity of natural languages is discussed at different levels of analysis: general similarity, and complexity of the morphological and syntactic representation. Languages encode meaning at different levels of structure. Some languages have complex morphology with a more flexible word order, whereas in other languages, the word

order carries the information of the constituents of the sentence. Solving this problem benefits, for example, machine translation, where these differences can cause problems.

In this work, three different unsupervised methods are used to analyze the similarity and differences of 21 European languages. Two of these methods are based on approximating Kolmogorov complexity (Kolmogorov, 1998; Li and Vitanyi, 1997) with compression (Juola et al., 1998; Juola, 2005, 2008; Li et al., 2004). The third method used is an unsupervised method of morphology induction called Morfessor (Creutz and Lagus, 2002, 2007), based on Minimum Description Length (MDL) (Rissanen, 1978), which is also a special case of Kolmogorov complexity.

1.2.2 Language as a resource: Distributional modeling of word meaning

In the second part, the focus moves to the level of individual words in a single language or a pair of languages. The majority of this dissertation concentrates on providing a computational representation for the *meaning* of words. For this purpose, the large amounts of text in electronic form are used as data, using the insight that words that are somehow similar, occur in similar company (Firth, 1957). This similarity of context can be turned into a vector space model, in which semantically related words are close, and unrelated words are distant (Schütze, 1993).

The vector space models are built based on large text corpora, and they are further compared to and analyzed with human similarity judgment data both in a single language (English) case, and in a simple bilingual (English-Finnish) setting. The latter setting is also useful contribution for machine translation, as it demonstrates how multilingual semantic representations can be generated. In this dissertation, different dimensionality reduction and visualization methods are used, such as Singular Value Decomposition (SVD), Principal Component Analysis (PCA) (Pearson, 1901; Hotelling, 1933), Independent Component Analysis (ICA) (Comon, 1994; Hyvärinen et al., 2001), Neighbor Retrieval Visualizer (NeRV) (Venna and Kaski, 2007; Venna et al., 2010), and the Self-Organizing Map (SOM) (Kohonen, 2001).

Most importantly, ICA applied to a word vector space model is shown to be able to find meaningful semantic components that correspond to human category judgments. Further, the performance of ICA is compared to a probabilistic method, Latent Dirichlet Allocation (LDA) (Blei et al.,

2003). Furthermore, the overlap of human category judgments and the word sets found with unsupervised methods is analyzed in detail: What kind of categories are found and why? What kind of categories are not found? And what kind of information do the unsupervised methods find beyond the evaluation labels?

1.2.3 Language of an individual and the community: Modeling the emergence of a shared vocabulary in a community

The origins of human languages are hidden in prehistory. The hypotheses of the different conditions that affect the language emergence can be tested through simulation (Cangelosi and Parisi, 2002). In the last part of this dissertation, a multi-agent simulation model for shared vocabulary emergence is introduced. In contrast to earlier parts of this dissertation, the language in the simulation is an artificial one and emerges in the course of simulation through agent interactions.

Another significant difference is at the representational level. In the distributional models, the meaning of a word is grounded only through its links to other words, whereas the words in the vocabularies of the agents are grounded through a mediating conceptual level to experiences in the simulated world, which satisfies the physical symbol grounding problem (Vogt, 2002).

The simulation model consists of two separate parts. First, it contains a formal model and a computational realization for an individual. The SOM is used as a model for the semantic memory, allowing concepts that are fuzzy and continuous. It is important to note that the conceptual representations of each agent are private and subjective, contrary to many models, where the agents share the common concepts. Second, a formal model for interaction between the individuals is defined. It is realized using the language game approach using naming games and two different alternative decision rules. As a result of the simulations, a shared vocabulary emerges in the population, despite the private conceptual representations of the agents.

1.3 Summary of publications and author's contributions

Publication I introduces a method for analyzing the complexity of the syntactic and morphological characteristic of multiple languages, which is

useful, for example, in machine translation. The methods use Kolmogorov complexity, which is approximated by compression. In addition, the morphological level of the languages was analyzed with an unsupervised morphology induction method, Morfessor. This was the first time Morfessor was applied to many of these languages. The present author took part in defining the problem setting, performed morphology induction analysis with Morfessor and took part in writing the article. In addition, some further evaluation was carried out for this dissertation by the present author. Further, while reviewing the article for this dissertation, the present author discovered an error in one of the experiments of Publication I, and carried out new experiments to replace the erroneous results. The new results are reported in this dissertation.

Publication II describes an approach, in which ICA is used in a bi-lingual setting using aligned sentences in English and Finnish. The authors jointly defined the problem. The present author evaluated the results and wrote the article together with the other author of the article.

In Publication III, visualization of adjectives using PCA, NeRV, and SOM are compared in a case of an under-studied part of speech, the adjectives. Antonym pairs were used in evaluation. The present author defined the problem setting jointly with the other authors, carried out most of the experiments and evaluation, and wrote a large part of the article.

In Publication IV, it is shown that ICA is able to find meaningful structure from the data that corresponds to category judgments given by human subjects. In addition, the article presents feature selection experiments for analyzing which features best separate a given category from other categories. The present author defined the problem setting jointly with the other authors, designed the evaluation experiments, and carried out the evaluation and analysis for the syntactic and semantic tasks, as well as wrote large parts of the article.

Publication V continues the semantic analysis in an ICA task with a test set that contains different relations within the categories. Furthermore, an analysis of different types of categories and relations between the categories of the test set were also conducted. In addition, the use of the SOM is demonstrated in a data exploration and visualization task. The authors defined the problem setting jointly, and the present author carried out the evaluation procedure and the experiments, as well as wrote most of the article.

Publication VI continues the semantic vector space analysis with ICA,

and compares the ICA results to results obtained with a probabilistic method, LDA. The analyses are carried out rigorously for two separate semantic category test sets. In addition, an analysis of the differences between categories in the test set is carried out, both visualizing the ICA and LDA results in an illustrative way, and using the SOM in visualization. In addition, a method for finding frequent sets of words from the ICA and LDA analyses is introduced, and a preliminary qualitative analysis of the coverage of the pre-existing labels is carried out. The authors defined the problem setting jointly. The present author defined the details of the evaluation and analyses, carried out the experiments and wrote the majority of the article.

Publication VII introduces a multi-agent simulation framework to model the emergence of communication in a population of agents. The SOM is used as a model of the agents' conceptual memory. The present author defined the problem setting with the other authors of the article and built the multi-agent simulation framework, carried out the experiments and evaluation, and wrote most of the article.

Publication VIII introduces a Bayesian-type selection process for learning in the language game setup. The present author was responsible for building the model jointly with the other authors, doing the experimental work, and most of the evaluation. The article was jointly written by the authors.

Publication IX proposes a theoretical framework for modeling communication between agents with separate conceptual models of their context. The present author took part in defining the problem setting and the theoretical framework, and in writing parts of the article related to language game research.

1.4 Structure of the thesis

This book covers a large number of different topics related to computational modeling of language and meaning. Due to the multidisciplinary nature of this work, the literature review is divided into three chapters ranging from linguistics and cognitive sciences to modeling and computational methodology used in this thesis. Chapter 2 provides the reader insight on language, meaning and theories on concept representation. In Chapter 3, issues related to modeling are discussed, focusing on modeling language and meaning. Chapter 4 contains the methodological building

blocks used in this dissertation.

Chapters 5 through 7 present the contributions of this dissertation. First, unsupervised approaches for analyzing complexity of languages are introduced in Chapter 5. The distributional models of word meaning are the topic of Chapter 6, and a multi-agent simulation model of shared vocabulary emergence is presented in Chapter 7. Finally, Chapter 8 summarizes the work.

2. Languages, words and meaning

This chapter provides an introduction to the characteristics of languages and a brief introduction to the levels of linguistic analysis. Further on, the meaning of words is discussed in relevant detail - from what meaning is, to the relation between referents in the world, the concepts or categories, and the words. Then, the discussion continues to introduce some theories of meaning representation, based on what is known of human category and concept processing.

2.1 Language

It is often said that the effortless use of language for communication sets us humans apart from other species. This ability is often considered to be the hallmark of intelligence (Pfeifer and Scheier, 1999). To communicate is not the only purpose of the language, though. Language serves seven main functions (Finch, 2003). These are i) to release nervous/physical energy, ii) for purposes of sociability, iii) to provide a record, iv) to identify and classify things, v) to be an instrument of thought, vi) to communicate and vii) to give delight. In this thesis, we concentrate on the purposes of identification, classification and communication.

A language can be viewed as a complex symbol system, structured in different levels. Languages exist in written, spoken and signed form. The symbols in a language carry meaning, agreed upon in a community of the users of the language. Languages can be divided along two distinctions. They can be either natural or artificial; and emerged or designed, see Table 2.1 for examples. Artificial languages include programming languages that are designed, but an artificial language could also emerge in a community of artificial agents.

Languages are made up of symbols, or words. A word is a linguistic unit,

Table 2.1. Types of languages

	Emerged	Designed
Natural	French, Finnish	Esperanto
Artificial	Emerged agent language	Perl, C, ...

which is a sound or a combination of sounds or the equivalent letters in writing, which communicates a meaning. Thus, no combination of sounds or letters is a word as such, unless it is connected to a *sense*. In the following, when a concept or a word meaning is referred, it is written in italics: *dog*. When the sequence of letters is referred, the term *form* is used, and it is written in single quotes: 'dog'. Most of this thesis concentrates on the relation between individual words and their meaning. In corpus work, two terms, *type* and *token* are often used. Type refers to the unique word forms such as 'cat' and 'hat' in the corpus. Tokens are instances of the type in the corpus, that is, word forms occurring in the running text.

2.1.1 Statistical properties of languages

In most natural languages, the distribution of the occurrence of the symbols follows an exponential distribution, also called Zipf's law: the frequency of a word is inversely proportional to its rank (Zipf, 1949; Manning and Schütze, 1999). This means that the most frequent word will occur approximately twice as often as the second most frequent word, three times as often as the third most frequent word etc. The implication is that while the most common words are very frequent, there is also a large number of words that are rare. As an example, let us consider the Grimm fairy tales[1]. The corpus consists of a total of 200 000 tokens. By counting how many times each unique word occurs, a vocabulary of 8 500 types is obtained, including punctuation marks. The most common type in the vocabulary is 'the' which appears almost 17 800 times. In contrast, there are also over 3 000 types that appear only once.

2.1.2 Language families

Natural languages spoken by people all over the world are numerous. While many languages have become extinct, there are over 7 000 lan-

[1]Available online at `http://www.gutenberg.org/files/2591/2591-_h/2591-_h.htm` Accessed September 23, 2013.

Table 2.2. The language families of the EU languages in 2006 (Katzner, 2002)

Family	Sub-family	language
Indo-European	Germanic (ge)	Danish (da)
		Dutch (nl)
		English (en)
		German (de)
		Swedish (sv)
	Romance (ro)	French (fr)
		Italian (it)
		Portuguese (pt)
		Spanish (es)
	Slavic (sl)	Czech (cs)
		Polish (po)
		Slovak (sk)
		Slovene (sl)
	Baltic (ba)	Latvian (lv)
		Lithuanian (lt)
	Hellenic (ie)	Greek (el)
	Celtic (ke)	Irish (ga)
Uralic	Finno-Ugric (fu)	Estonian (et)
		Finnish (fi)
		Hungarian(hu)
Semitic (se)		Maltese (mt)

guages still spoken in the world (Lewis et al., 2013). Languages evolve, and sometimes new languages are born. Linguists have found it useful to categorize languages into families to ease their description. In computational modeling, it also often makes sense to be aware of the language family structure. A model of a language that has been tested only on one language family, might not be applicable to other language types.

Publication I concentrates on the official languages of European Union (EU) in 2006. At the time of the publication, the 21 official EU languages were Czech, Danish, Dutch, English, Estonian, Finnish, French, German, Greek, Hungarian, Irish, Italian, Latvian, Lithuanian, Maltese, Polish, Portuguese, Slovak, Slovenian, Spanish and Swedish. Most of them belong to the Indo-European family of languages, as shown in Table 2.2. In Publication I, these categorizations into language families are compared to findings obtained with unsupervised methods.

2.1.3 Levels of linguistic analysis

Language is a complex system with some internal structure. To make sense in the analysis, linguistic analysis often focuses on a perceived level in that structure. In the following, the levels of linguistic structure are covered briefly.

Phonology

Phonology is the level of the structure of the sounds organized into words (Clark and Yallop, 1990). The *phoneme* is the basic unit to build words, usually defined as the smallest unit in language that can cause a change in meaning, for example, [ɪ] and [æ] in English words 'hit' and 'hat'. A phoneme is an abstraction that can consist of different produced sounds, or allophones. Phonology is not further discussed as this work concentrates on written language.

Morphology

Morphology is the study of the structure of words and their variation. The words are built of *morphemes*, which are the smallest meaning-carrying units in a language (Matthews, 1991). A linguistic *morph* is the surface form of a morpheme, a phonetic realization of a morpheme. The complexity of the morphological structure varies in different languages. Meaning can be encoded either as separate words or added as a prefix or a suffix; or as a pattern in languages such as Arabic or Hebrew. In morphologically complex languages, words may consist of several morphemes, each with their own meaning. Finnish is an example of such an *agglutinative* language, where parts of words are 'glued' together. An often used example is the Finnish complex compound noun 'kahvinjuojallekin' which separates into six morphemes 'kahvi+n+juo+ja+lle+kin' ('also for the coffee drinker') (Creutz and Lagus, 2002). Vietnamese is an example of an isolating language, with little morphology.

The complexity of the morphological level of 21 European languages is studied in Publication I. In addition to expert studies of morphology, it is possible to extract morphological structure from data in an unsupervised way. A particular method, Morfessor (Creutz and Lagus, 2002), is discussed in more detail in Chapter 4.

Syntax

Syntax is the level of organization of words into larger constructs: phrases or sentences, and the study of the rules governing that structure. Again, the syntactic structure varies from language to language. Some languages employ a strict word order, which is needed to mark the function of the constituents of a phrase, whereas other languages are more flexible in this regard. A general rule is that in languages in which the words contain grammatical information in the form of inflection, the word order is more flexible, even though there is usually a preferred word order (Com-

rie, 1993). The languages are often characterized by the order in which the Subject (S), Verb (V) and Object (O) appear in a sentence. All orderings are possible, but SOV, SVO and VSO are most frequent in that order (Greenberg, 1975). The remaining three orderings are rare but not impossible. It is hypothesized that SOV is the original word order (Gell-Mann and Ruhlen, 2011) from which other word orders have developed. The differences in word order in different EU languages are studied in Publication I, in which the similarity and complexity of different levels of language are analyzed.

The words can be classified into categories based on whether the words behave in a syntactically similar way. These categories are called the grammatical or syntactic categories, or Parts of Speech (POS). Most typical grammatical categories are noun, verb and adjective. Typically, nouns denote people, animals, concepts and things, verbs describe action, and adjectives are used to describe properties. These classes are open: new words can come into these classes (Manning and Schütze, 1999). Other categories, like pronouns, determiners and prepositions are closed: they do not get any new members. In Natural Language Processing (NLP) applications, the categories are sometimes further divided into more specific categories, for example, making distinction between singular and plural nouns or proper nouns, or different cases of verbs. The oldest and probably most used resource is the Brown corpus (Francis and Kucera, 1964), which has 81 different grammatical category labels.

Semantics and pragmatics

In linguistic study, semantics is the study of the meaning in language. The elements may carry meaning at different levels: at the level of morphemes, individual words, phrases, or sentences. Traditionally, semantics has concentrated on the senses of the words that are seen as shared between the speakers of the language. Pragmatics is the study of the context which affects the interpretation of the words. Cognitive semantics differentiates from traditional semantics in a sense that it questions the existence of strict linguistic levels introduced above (Croft and Cruse, 2004). It takes into account the embodied self: the meaning must be grounded in bodily experiences. Thus, it does not fit into the classic division, but it also extends to the field of pragmatics, as the context needs to be taken into account as well.

2.1.4 Meaning and sense

In semantics, a distinction between conceptual and associative sense is sometimes made (Finch, 2003). Conceptual sense is assumed to be shared between the users of language: the 'objective' meaning, whereas the associative sense is based on our individual experiences. The associative sense is thus more subjective.

The meaning of word *sense* is almost synonymous to the meaning of *meaning*. In this dissertation, it is used to mean the different senses of a single word, whereas the word *meaning* is reserved for more general use. *Sense* can be defined as a general level of meaning, which is something the speakers of the language more or less share (Finch, 2003). This can be further defined as conceptual sense, for example, what a *man* and a *woman* mean. Further, words do get associations on top of the general meaning from social and cultural context, and these associations are called connotations.

One can note that words are polysemous by nature. Often a word form has different meanings, which are somewhat related, but mean different things in different contexts. As new words are invented constantly, we could have different words for each of these meanings, but it seems that it is more efficient for communication to have a smaller set of shared, ambiguous labels than a large set of unambiguous labels (Zipf, 1949).

2.2 The relationship between the world, meaning and words

As defined earlier, words communicate a meaning, and the string of letters does not mean anything without that link to the meaning. Words can be used to refer to, for example, objects, instances, and abstractions: something that lies outside the language system. The thing that is being referred to, whether it is an object, an instance, or an abstraction in the world, is called the *referent*. The following sections will discuss this relationship between words, their meanings, and their referents. Researchers in philosophy, psychology, cognitive science, and linguistics have differing viewpoints on whether an abstract level of representation is needed, what kind of representations there are, and how should the computational representations be realized. The summary of the viewpoints given here will not be in depth, but rather highlight the points relevant for this dissertation.

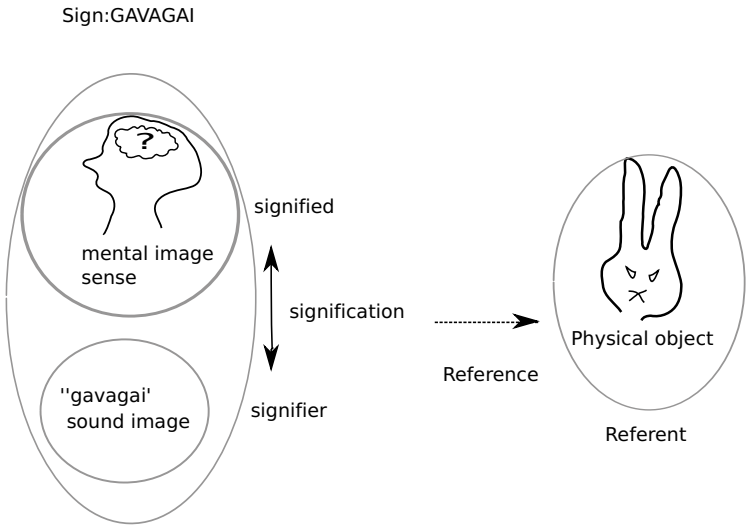


Figure 2.1. The dyadic sign of Saussure according to (Finch, 2003)

In semiotics, a sign is something that we humans assign a meaning to, which stands for 'more than itself' (Chandler, 2000). In the following, two models of signs to represent the relationship between words, meanings and referents will be introduced.

2.2.1 The Saussurean sign

The Saussurean model of a sign (Saussure, 1966) is divided into two parts: the *signifier* and the *signified*, with a process of signification between these two, illustrated in Figure 2.1. Saussure's viewpoint was a *structuralist* one, and it concentrates on the system of language itself, without reference to the world, and the theories consider the relations between the signs. Saussure's ideas were highly influential in the field of linguistics in general (Finch, 2003). He introduced the notion of *syntagmatic* and *paradigmatic* relations between words. Words are in *syntagmatic* relation, if they occur together in a phrase. In a *paradigmatic* relation, words can replace each other (Saussure, 1966). For example, the words in a phrase 'fox ran', the words 'fox' and 'ran' are in a syntagmatic relation, because they co-occur. However, the words 'fox' and 'cat' do not co-occur, but 'cat' can replace 'fox' in the 'fox ran' construction, thus 'cat' and 'fox' are in a paradigmatic relation.

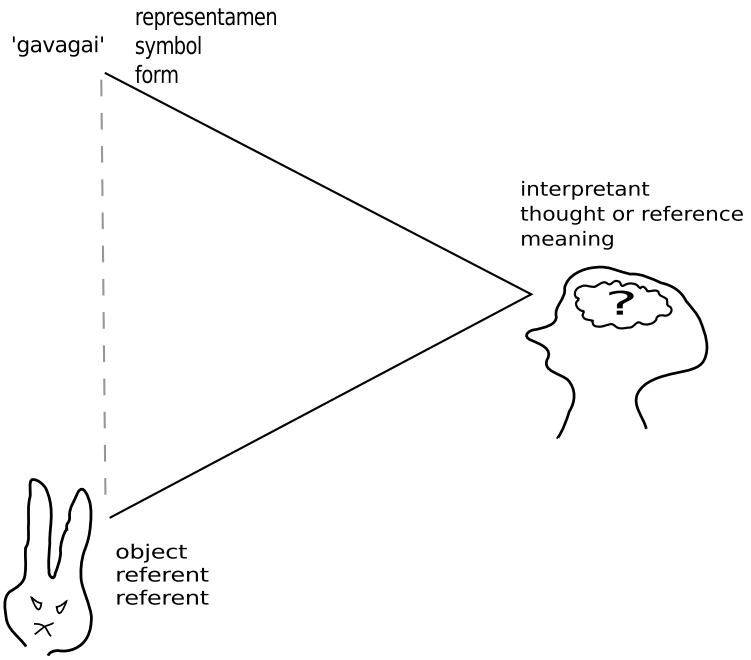


Figure 2.2. The semiotic triangle with the terminology used by Ogden and Richards (1972), Peirce (1931), and Steels and Kaplan (1999) and Vogt (2002)

2.2.2 The semiotic triangle

Another way to describe the relationship is the *semiotic triangle*. This model will also be used in this dissertation. Peirce (1931) defines a sign as "anything which determines something else (its interpretant) to refer to an object to which itself refers (its object) in the same way, the interpretant becoming in turn a sign".

In contrast to the Saussurean sign, the semiotic triangle has three parts as the referent is taken into account as well. The three corners of the triangle are the referent which is the instance or object or abstraction 'in the world', the representation of the referent and the symbol used to denote it. It is important to note that there is no direct link between the referent and the symbol, but all linking must go through the representation level.

There are several versions of this triangle in the literature. The ones most often quoted are probably Ogden and Richards (1972) and Peirce (1931) who used slightly different terminology. Peirce used the terms 'representamen', 'interpretant' and 'object', whereas Ogden and Richards use 'symbol', 'thought' or' reference', and 'referent'. In their work, Steels and Kaplan (1999) and Vogt (2002) use the terms 'form',' meaning' and 'referent'.

There is a field of research dedicated to study of the signs, semiotics. An in depth treatment of the signs is beyond the scope of this thesis. In this dissertation, the semiotic triangle is used as a practical model outlining the problem field.

2.3 Concepts and categories

The word *concept* will be used to describe the representational level between the referent and the form. This section will discuss the notion of concepts and categories and theories that have been built to represent them. Later on, the philosophical and psychological models presented here will be discussed in the context of computational models.

In general, concepts are seen as mental particulars in the field of linguistics and cognitive science. However, this is not agreed by all. For example, it is usual to think of concepts as abstract entities in the field of philosophy (Laurence and Margolis, 1999). Very often theories of concepts concentrate on the notion of lexical concepts, that is, concepts that correspond to lexical items in natural language. In this work, the word concept is taken as a means for specifying a relationship between the world and the language.

Another concept often used in this context is *category*. Category is a class or division of items which are regarded as having particular shared characteristics. The members of the category are usually considered equal in some sense. The division can be general or comprehensive (Rosch, 1978). In this dissertation, category names are written with ALL CAPITALS, for example, ANIMAL. In a certain way, categories and concepts are two sides of the same coin (Gärdenfors, 2000).

For an individual, it is useful in many ways to be able to distinguish between perceived items. This sorting to categories is often called *categorization*. It has a major role in perception, thinking and language, and it is probably significant in motor performance as well (Harnad, 1987).

Rosch (1978) presents two principles according to which the category systems function. These principles are 1) cognitive economy and 2) perceived world structure. The function of the category systems is to provide maximum information regarding the world with the least cognitive effort. In other words, categorization helps us to reduce and make sense of the information flow we constantly receive through our senses.

Categorical perception is a phenomenon that helps us to sort our sen-

sory input to invariant categories: differences within a category are perceived smaller, and differences between categories are magnified. This phenomenon has been demonstrated, e.g., for speech (Harnad, 1987). Categorical perception consists also of, for example, mapping different views of a face into a common identity, or mapping distinct speech tokens into a same phonetic percept. Categorical perception might be viewed as a proof for the existence of some kind of an abstract level of representation (Rosch, 1978).

2.4 Theories of concept representation

There are many theories about the structure of the representations and their computational realizations. These models tend to address different kinds of phenomena related to the concepts. In the following, some of them are covered in a general level. This section relies heavily on Murphy (2004). What is common to these theories is that none of them is all-encompassing, but they cover and explain a part of the phenomenon. In fact, some suggest that humans have separate concepts of the three kinds: prototypes, exemplars and theories (Machery, 2009). The models are presented here as an underlying theoretical foundation on which the models discussed in the later chapters are based. They will be revisited in later chapters where computational models are discussed.

2.4.1 The Classical View

The classical view was to see the concept as a set of definitions (Laurence and Margolis, 1999) necessary and sufficient for something to be of that concept. According to this view, for example, a concept for 'bird' could include the following definitions: 'has wings', 'can fly', 'lays eggs', etc. and filling these conditions would be necessary and sufficient for something to be a bird. This kind of notion on concepts has a long history in philosophy (e.g., Locke, 1690/1975).

The classical view was challenged, when it was noticed that many concepts exhibit typicality effects (Laurence and Margolis, 1999). The typicality effects mean that humans tend to judge some members of the category more typical than others. A *sparrow* is judged to be a more typical example of the category BIRD than a *penguin* (Rosch, 1978). In addition, the borders of the concepts may be fuzzy. The classical view cannot account

for these effects. According to the classical view, any concept that satisfies the necessary and sufficient conditions is an equally good member of that category. Other phenomena, such as family resemblance (Wittgenstein, 1963), do not fit into the classical view either. Currently the classical theory is not considered valid by most researchers (Murphy, 2004).

2.4.2 Prototypes

Typicality effects led researchers to hypothesize that most lexical concepts are represented as prototypes, which encode a statistical analysis of those properties the members of a given category tend to have (Laurence and Margolis, 1999). This view is also controversial, and it has been argued that the existence of prototypes tells us nothing about concepts, since well defined concepts also exhibit typicality effects. It also seems that there are some concepts for which people fail to represent any central tendencies at all (Laurence and Margolis, 1999). Rosch (1978) writes that empirical findings of prototypicality effects have been confused with theories of processing: they seem only to constrain, but not specify the representation and process models. Prototypical members tend not to be those that are frequent, but those which have many prototypical features or qualities associated to the members of a given category (Murphy, 2004).

2.4.3 Exemplars

An alternative to a representation that encompasses an entire concept as some kind of a summary representation is that the concepts are stored as exemplars of encountered concepts (Medin and Schaffer, 1978). This would mean that a concept *dog* would consist of all those dogs the person remembers, either consciously or unconsciously more or less clearly—some exemplars would be fuzzy due to forgetting (Murphy, 2004). The similarity of a new item would be then compared to remembered category items, and typical items would be categorized faster as they are similar to a large number of category members. Exemplar theory can explain many phenomena related to concept learning. For example, when encountering a new concept for the very few first times, the concept consists of the encountered example. It is then a matter of debate whether an aggregate concept from exemplars is formed or not.

2.4.4 The knowledge-based approach

Another theory concentrates on the relations between concepts. This theory has many names: knowledge or explanation based theory or theory theory. The basic idea of this theory is that concepts are an integral part of our knowledge about our world, and cannot be considered in isolation (Murphy, 2004). Humans also possess mental theories about the world, from early on, although the theories can and often are inaccurate. This theory cannot cover the entirety of the concept learning, as many concepts cannot be based on previous knowledge only, and thus the theory needs a separate or integrated model of concept learning through experiences.

3. Computational modeling of language and concepts

A variety of computational models are used in this thesis. The models are based on human cognitive abilities on meaning representation and linguistic skills. This chapter first provides a general discussion of the philosophy of computational modeling and simulation and introduces the concept of emergence, which is then followed by a general introduction of the models used in this dissertation. The methodological details used to build those models are then explained in the following chapters.

3.1 The purpose of modeling

This thesis concentrates on computational models used to represent meaning. Thus, it is important to discuss the concept of modeling and its purpose. In general, models provide a coherent framework for interpreting data, highlight basic concepts, uncover new phenomena, or identify components of a system. They can link levels of detail and known phenomena to that still unknown, and inform experimental design. They also expand the range of questions that can be meaningfully asked and can be used to screen out unpromising hypotheses (National Research Council (US), 2005).

In the context of cognitive science, Pfeifer and Scheier (1999) give two different purposes for a model. From one perspective, the purpose of a model is to better understand a certain phenomenon. From another perspective, also called the engineering perspective, the purpose of modeling is to build systems that simply work and serve practical purposes. Seen this way, the intent behind modeling affects the selection choices—if the purpose is to build a system that works, it does not need to mimic the underlying components very faithfully. On the other hand, if we want to understand the phenomenon, this is very important. In cognitive science,

models are usually used for three different purposes (Morse and Ziemke, 2008): First, modeling can be used for testing whether a particular model is sufficient: i.e. whether a model is able to produce data that matches the data from a real phenomenon. Second, as a sort of Occam's Razor, questioning the necessity of a given theory. If a simpler model can produce certain results, it may sometimes be concluded that a more complex model is not needed. Third, modeling is used increasingly as an explorative tool for the interactive potential of a situation. McClelland (2009) also discusses the role of modeling in cognitive science and sees the models as "explorations of ideas about the nature of cognitive processes," in which simplification is essential to see the ideas more clearly. In addition to explorative models in cognition, there is a whole field of explorative data analysis. The results of the exploration can be then compared and contrasted to models and evaluation sets built on expert knowledge. This topic is discussed more in Chapter 4 in the context of vector space models.

Pfeifer and Scheier (1999) distinguish between analytic and synthetic approach for modeling. In an analytic approach, the experiments are done on an existing system and a model is built to predict the outcome of the future experiments. To solve a research question by the synthetic approach means that an artificial system that reproduces certain characteristics of a natural system is created. The focus is on trying to reproduce the internal mechanisms that have led to the particular results. In this dissertation, both approaches are used. The analytical approach is used when meaning representations are created from language data. A synthetic modeling approach is then used when language learning is modeled in a community of simulated agents.

3.1.1 Simulation

Simulation is imitation of a real process or a system over time, usually by means of computer programming. We can further distinguish agent simulation, where the modeled programs are agents: autonomous, functioning 'individuals' able to interact within the (simulated) world. Simulation models can be used in cognitive science in the three different ways as explained earlier: to test sufficiency or necessity, and in exploration, which is defined as investigation of agent and environment embedding (Morse and Ziemke, 2008).

Synthetic modeling can be realized in either robots or in a simulation. In robotic experiments, the physical world can be used, whether it is mod-

eling dynamics, sensors or motor control. In simulations, realistic physical modeling is computationally expensive and difficult. Simulations suit better in population modeling, as copies of agents can be reproduced infinitely. In addition, simulations can also be run in parallel, tweaking parameters is easier in simulations, and they rarely need constant supervision from the experimenter. Currently evolution can only be simulated. In the case of robots, only the controller programs can be evolved. In short, simulation is "fast, cheap and flexible" (Pfeifer and Scheier, 1999). Simulation models are used in Publications VII and VIII of this thesis.

In this thesis, the focus is on simulations of *agents*, which are able to perform some tasks in a given simulated *environment* which can change due to the actions of the agent. More specifically, the model is a *multi-agent simulation model*, where different sizes of *populations* of agents are used. Yet another interaction is introduced into the model: the interaction *between* the agents. This means that in a multi-agent simulation, we need at least a) the environment in which the agents are situated; b) the model of the agent and its attributes; c) the model of agents' interactions with the world; and d) the model(s) of interaction between agents.

3.1.2 The levels of abstraction

In any model, abstractions must be made. One purpose of the model is to simplify the phenomenon it attempts to describe, describing only the important components. A core problem in modeling is to decide which variables to include in the model, and which to exclude (Marechal and Thomas, 2007), that is, the level of abstraction. For example, when modeling navigation behavior of ants, one can assume that wheels instead of legs might not affect the behavior (Pfeifer and Scheier, 1999).

Marr and Poggio (1977) identify three levels of representation: a) physical realization, b) the algorithm and c) the overall computation. At the algorithmic level, there are several different ways to implement an algorithm, which in turn may constrain the physical realization. At the top level is the description of the actual computation: the description of the problem we want to solve. These ideas apply to modeling in cognitive science as well (Morse and Ziemke, 2008). For example, there will always be multiple possible algorithms that produce the same data. Top-down analysis cannot produce an accurate account of mechanisms genuinely in use, but a bottom-up approach could be possible, if it is accompanied in part by the empirical data it accounts for.

3.1.3 Emergence

Emergence is a term often used in philosophy, and in the fields of complex systems, artificial intelligence and unsupervised learning. The term is also used to describe different phenomena. Goldstein (1999) gives a general definition of emergence:

> "Emergence, [...] refers to the arising of novel and coherent structures, patterns, and properties during the process of self-organization in complex systems. Emergent phenomena are conceptualized as occurring on the macro level, in contrast to the micro-level components and processes out of which they arise."

Chalmers (2006) points out that the term is actually used to describe two different phenomena: strong and weak emergence. A strongly emergent phenomenon is not deducible even in principle from the truths in the low-level domain - and it is the notion of emergence most common in philosophical discussions. A weakly emergent phenomenon, on the other hand, arises from the low-level domain, but the higher-level behavior is *unexpected*. Pfeifer and Scheier (1999) note that in engineering, *emergence* is often used to describe a) behaviors that are surprising or not fully understood; b) property of a system not contained in any of its parts or c) behavior resulting from agent-environment interaction that is not preprogrammed.

In this thesis, we will not concentrate on philosophical definitions but rather discuss results in the fields of artificial intelligence and language modeling, where the term is often used, and unless otherwise stated, *weak emergence* is meant by the term *emergence*.

In the domain of language, language learning as emergent phenomenon is contrasted to the nativist approach which presupposes genetically-wired language acquisition devices (Croft and Cruse, 2004). For example, neural network models have been used to show that it is possible to model language learning (MacWhinney, 1998).

The evolution and emergence of a simple language using computer simulation has been studied extensively. See, for example, Steels (1996) and Cangelosi and Parisi (1998) for early works. In this setting, agents communicate with signals and play communication games. The signals are selected according to some criteria, and it is possible to study in this set-

ting the emergence of shared vocabularies. The language game simulations will be further discussed later in this chapter in Section 3.4.1, and in Chapter 7.

In the context of unsupervised learning, emergence has been also used to describe 'emergent features', that is, latent features or descriptions which emerge from the data when an unsupervised method, such as Independent Component Analysis is used (Honkela et al., 2010). This topic is further discussed in Chapters 4 and 6.

3.2 Similarity and distributional models of word meaning

The notion of *similarity* is very central in this dissertation, and all the computational models used here translate the similarity of items as proximity in some spatial representation. Sahlgren (2006) notes that this is very intuitive to humans, as *similarity is proximity* is one of the very basic metaphors humans use (as identified by Lakoff and Johnson, 1999).

Similarity as proximity is used in Publication I of this thesis, where different properties of languages are studied, and languages are classified similar or dissimilar based on how they are represented in different levels of linguistic analysis. This notion is of course more simple, when the representation of a whole language is reduced to one or two dimensions. The word vector space model, on the other hand is a complex, high-dimensional representation. Further, in Publications II–V, the same notion of similarity is used in the vector space models, see Sections 3.2.2 and 4.4 and Chapter 6; and in Publications VII and VIII for the geometric representation for a simulated agent's semantic memory, where occurrences that are mapped close to each other in the semantic memory are also semantically similar. See Section 3.3.4 and Chapter 7.

3.2.1 Similarity of word meanings

Earlier in this dissertation, words were defined as symbols associated with a meaning. This means that when discussing word *similarity*, we mean the similarity of the underlying meanings. Similarity of words cannot thus be measured by the similarity of surface forms. For example, edit distance between text strings can be measured as the number of deletions, insertions or substitutions needed to arrive from one word to another, which for 'at' and 'hat' would be 1, but this does not tell us any-

thing about the similarity of the *meaning* of these words. For that, we need to represent the meaning in another way. We will discuss a solution, namely the vector space models in later parts of this dissertation.

What is word similarity, then? Budanitsky and Hirst (2006) differentiate between semantic *similarity* and *relatedness*, of which the latter is more general term: words that are dissimilar can also be related. Meronymy and antonymy are examples of such relations. They also make a distinction between the word similarity of the underlying concepts and computational approaches such as Dagan et al. (1999) in which the term 'word similarity' is used to describe the similarity of the distributional properties of the words.

Several test sets based on human similarity judgments have been built by simply asking test subjects which words they find similar or related. The basic practice in evaluating the computational models is to compare the results to these similarity judgments. This kind of evaluation sets are described in detail in Chapter 6.

3.2.2 The vector space model of words

In this section, the purpose and background assumptions of a word vector space model are discussed in a fairly general level. The technical details of building such a model and the experiments carried out in the publications that contain the major contributions of this dissertation are discussed in Chapters 4 and 6.

The distributional models are all based on a general hypothesis that statistical patterns of human word usage can be used to find out what people mean (Turney and Pantel, 2010). Originally, the Vector Space Model (VSM) was developed to represent documents as vectors for Information Retrieval (IR), that is, finding relevant documents for a given query from a collection of documents (Salton and Buckley, 1988). Vector space models of words, the focus in this thesis, carry many features of the document models, beginning from the idea that words, like documents, can be represented by other words that occur in a certain context around them. The distributional hypothesis means that words that occur in similar contexts, that is, words that are distributionally similar, are also more similar in meaning. When reviewing the history of such models, Harris (1954) and Firth (1957) are often quoted. Sahlgren (2008) gives an analysis of the history of the distributional models, of which the gist can be compressed into a following quotation from Rubenstein and Goodenough (1965): "words

which are similar in meaning occur in similar contexts". Sahlgren (2006) also ties the vector space models of words into the structuralist linguistic tradition.

Erk (2012) notes that the terms 'distributional model' and 'vector space model' are sometimes used synonymously. She makes a distinction between the two: The former are built explicitly from distributional information of contexts, whereas the latter are any high-dimensional vector representations, regardless of the origin of the features. In the models used in this thesis, both definitions apply.

Context can be defined in several different ways. The technical aspects of context selection will be discussed later along with many other practical choices that need to be made. When words are represented as vectors of features, we obtain a spatial representation of word meaning (Sahlgren, 2006). In a space such as this, the similarity is measurable as a distance in the space using general distance measures.

Thus, we can build co-occurrence count representations for words by looking at the words in large text corpora, which are more and more available in electronic form. For such a model to be built, enough language data is needed to produce vectors that are statistically reliable representations.

Word vector space models are also called word space models (Schütze, 1993), or semantic spaces (Baroni and Lenci, 2011). The last term is also used in psychology (Finch, 2003), where the notion of semantic space is used as a metaphor without specifying the computational model.

In distributional models, the complex meaning of a word is substituted by the co-occurrence count representation. This is a large simplification, which leaves out all extralinguistic information, as meaning is only grounded through word use. Comparing to the discussion in Section 2.2, we see that this method is thus a structuralist one, corresponding to the Saussurean sign, or two out of three corners of the semiotic triangle. Yet, even with this simplification, the model works reasonably well.

Word vector space models have been criticized for a too broad notion of semantic similarity, which limits the applicability of the models, as the different types of similarity (antonyms, synonyms, hyponyms, etc...) cannot be distinguished (Pado and Lapata, 2003). Sahlgren (2008) notes though that the corpus-based distributional model is descriptive, and a broad notion of semantic similarity works well as humans make judgments about semantic similarity easily.

Vector space models, based only on language data and thus ungrounded,

are certainly not alone sufficient models for all of the human language processing capabilities. However, Landauer and Dumais (1997) suggest that humans learn a large part of their vocabulary from text, and vector space methods that use co-occurrence data replicate this phenomenon and acquire a knowledge of the vocabulary of English in a same level as school children. Bullinaria and Levy (2007) propose that statistical representations could form a foundation for learning of the semantic representations. They suggest that while a statistical approach to language learning is not sufficient alone, humans do take advantage of such methods.

3.2.3 Structured models for lexical semantics

There is naturally more to representing meaning of words than simply representing individual words with vector space models. These models are left outside the scope of this thesis, but a brief account is given on those models as well. Quite a few of the models concentrate on describing the structure of domain knowledge representation of a complex concept, or its relation with other concepts, resembling knowledge-based concept representations.

The vector space models introduced here concentrate on single words, but there is a growing research effort on building compositional models to represent, for example, word pairs. See, Turney and Pantel (2010) and Clark (forthcoming) for review.

Structured approaches, such as ontologies are useful when we want to describe relations. They are usually built by experts and for each language separately, and as such, building them is costly. A large resource of this type is the lexical database WordNet (Miller, 1995), available in many languages. Wordnets are large linked thesaurus-type networks, where links can represent different relation types. Similarly, frames (Minsky, 1975) are knowledge (or data) structures intended to describe typical situations extended to frame semantics (Fillmore, 1976) relating linguistic semantics to general knowledge needed to understand that word. A computational resource of such a type is the FrameNet (Baker et al., 1998). Bayesian models of cognition combine statistical methods and structured knowledge resources (Tenenbaum, 1999; Tenenbaum and Griffiths, 2001), for examples on decision making.

3.3 Modeling linguistic cognition

The second large part of this dissertation consists of agent simulation models of language emergence. This part contains models for concept acquisition and representation and a simulation model of communication, which leads to the emergence of a shared vocabulary in a population.

The simulation model developed leans heavily on the research on computational cognitive science and Artificial Intelligence (AI), for which a brief historical account is given.

3.3.1 Traditional symbolic AI

The early AI research concentrated on attempts to replicate the human level of intelligence on a machine (Brooks, 1991). Obviously, this failed. Research has since then concentrated on demonstrating isolated aspects related to cognition and intelligence. The early approach concentrated on *problem solving*, and saw that as a hallmark of intelligence.

Traditional AI was symbolic in nature. At that point a cognitive agent was viewed as a kind of logic machine, which operated on symbols that form the concepts (Lakoff, 1987). The symbols were seen as the necessary and sufficient basis for general intelligent action (Newell and Simon, 1976). The intelligent behavior consisted of manipulating these symbols according to some rules. The core of intelligence was seen as problem solving: A physical symbol system would show its intelligence in problem solving by search - that is, by generating and progressively modifying symbol structures until it produces a solution structure. Such a symbol system was then thought to be realizable with a universal machine (Newell and Simon, 1976).

3.3.2 The symbol grounding problem and embodied cognitive science

The early symbolic systems had problems: they could not function in a changing environment or learn. In addition, it was questioned where the mental symbols get their meaning. This question is the *symbol grounding problem* (Harnad, 1990, p. 335):

> "How can the semantic interpretation of a formal symbol system be made intrinsic to the system, rather than just parasitic on the meanings in our heads?

How can the meanings of the meaningless symbol tokens, manipulated solely on the basis of their (arbitrary) shapes, be grounded in anything but other meaningless symbols?"

Harnad's solution to the symbol grounding problem was to ground symbolic representations bottom-up from two kinds of non-symbolic representations: 1) iconic representations that are analogs of proximal sensory projections and 2) categorical representations that are learned by innate feature detectors picking up the invariant features of object and event categories from the sensory projections. The elementary symbols would then be names for these categories and higher-order symbolic representations would be grounded in the elementary symbols.

Steels and Vogt (1997) offer the semiotic triangle (refer back to Section 2.2) and its application in an agent simulation or a robotic experiment as a solution of solving the symbol grounding problem. This notion for the triangular structure of a sign—or the relationship between the world, the meaning representations and the symbols to denote them—is important throughout this thesis. The research questions address different aspects surrounding the semiotic triangle and new fields emerged to tackle these questions using different methods.

The viewpoint of the embodied cognitive science is that embodied cognitive models structure thought and they are used in forming categories and in reasoning. The cognitive models characterize the concepts, which are used via the embodiment of the models. A further note on the embodiment is that most cognitive models are embodied with respect to use. Abstract conceptual structures are indirectly meaningful: they are understood because of their systematic relationship to directly meaningful structures (Lakoff, 1987).

Of course, the embodiment approach as such does not provide help in finding out how one gets from the continuous sensory signals to the symbolic level of words, but it provides a link between the meaning creation and word acquisition.

3.3.3 Learning: Neural network based approaches

Learning was another important aspect neglected by the early AI models. Connectionist or neural network models brought learning and development of cognitive properties more into focus, and the models have

been able to give accounts for a range of different developmental phenomena, such as infant category development and language acquisition and reasoning in children (Thomas and McClelland, 2008). The models were based on the principle that the information processing properties of neural systems, such as parallel processing, should be taken into account when designing the models (Thomas and McClelland, 2008). The level of explanation is different in the neural network models: there the emphasis was on developing Parallel Distributed Processing (PDP) models (Rumelhart et al., 1986; Haykin, 1999).

Artificial neural networks are massively parallel distributed systems, made of simple interconnected processing units (or neurons), which send each other excitatory and inhibitory signals. The system is adaptive: the network acquires knowledge from the environment through a learning process and as such it is a bottom-up process contrary to a symbolic top-down approach. The knowledge is stored in the connection weights between the interconnected units. As nonlinear models, they are suitable for modeling inherently nonlinear phenomena (Haykin, 1999). In neural network models, multiple sources of information can be considered simultaneously, representations are spread across multiple processing units in parallel, the representations are graded, context sensitive and emergent, and computation is similarity-based, but can produce rule-following behavior (Thomas and McClelland, 2008).

Over the years many different variations of neural networks for different purposes have been developed. As a text book account, Haykin (1999) can, for example, be used. The only (artificial) neural network model used in this thesis is the Self-Organizing Map (Kohonen, 2001), which is introduced in Chapter 4. As an account of the contribution of neural network (or connectionist) models for cognition, one can, for example, use Thomas and McClelland (2008).

3.3.4 Conceptual spaces

Conceptual spaces theory (Gärdenfors, 2000) is a theory of geometric meaning representation, used in Publications VII and VIII and IX. The main idea of the theory is that concepts are modeled as geometrical areas in a multidimensional conceptual space rather than as symbols or activations between neurons. The conceptual spaces theory proposes a mediating level between sensory and symbolic levels. It provides a medium to get from the continuous space of sensory information to a higher conceptual

level, where regions in it could then be associated to discrete symbols.

A conceptual space is built upon geometrical structures based on a number of quality dimensions, which represent various qualities of objects. A conceptual space $\mathcal{C}$ consists of a class $D_1, D_2, \ldots, D_n$ of quality dimensions. A point in the space is represented by a vector $\mathbf{v} = [d_1, d_2, \ldots, d_n]$. Concepts are not independent of each other but can be structured into domains. For example, concepts for colors are in one domain, and spatial concepts are in another domain. The concepts are convex regions in the conceptual space spanned by the quality dimensions, and learned from a limited number of examples and by generalizing from them. Temperature, weight, brightness, and the spatial dimensions height, width and depth are examples of such quality dimensions, but the metrics in the dimensions need not to be absolute but can depend on the perceiver. The theory also suggests that despite the variations, scientific representations could be used in construction of an artificial system, where the input on different sensors is described in terms of scientifically modeled dimensions.

Conceptual spaces theory also incorporates the concept of *similarity is proximity* discussed in Section 3.2, which makes it computationally practical. The similarity of two objects can be defined as a distance between their representation points. This distance measure can then be used, for example, in categorization: A perceived item is mapped to the conceptual space, and it belongs to the category for which the prototype is closest to its representation in the conceptual space.

Gärdenfors (2000) proposes that certain neural network or statistical methods, such as Multi-Dimensional Scaling (MDS) and SOM could be used as a basis for a domain in a conceptual space. The SOM reduces the dimensionality of the data in a systematic and meaningful way, which can be seen as moving from sub-conceptual to conceptual level. This is the basis of the agent simulation experiments described in Chapter 7 and Publications VII and VIII.

3.4 Simulating vocabulary acquisition in a community

This section presents models related to language acquisition and communication relevant for this dissertation. Rather than assuming that a natural language is an innate biological system of humans, in this dissertation it is assumed that human languages are learned through a cultural process (Tomasello, 1999), bearing in mind that there are underlying biologi-

cal components that make language acquisition possible. For a discussion of biological adaptations required for cultural transmission of language to be possible, see Hurford (2003). For the purposes of modeling language emergence, it is assumed that language is a complex adaptive dynamical system (Steels, 1996).

Computer simulations provide a useful tool for studying questions related to language origins and evolution, a question which is sometimes dubbed as the hardest question in science (Christiansen and Kirby, 2002). As the beginning of human language use is hidden in prehistory and cannot be repeated, simulation models allow the testing of different theories concerning the mechanisms of language evolution.

The questions related to different issues in language emergence and evolution are numerous. They range from the question of why language has evolved, to how we are able to use symbols at all, and further to how a shared signaling using symbols in a population can emerge. Further, different properties of languages such as the compositionality of language or emergence of syntax are studied. Currently, there are several collections that cover the current research on the simulation on language evolution and emergence. See, for example, Lyon et al. (2007) for a review on the research on emergence of language; Kirby (2002) for a review on emergence of syntactic structures; and Cangelosi and Parisi (2002) for simulating language evolution.

Agent simulation systems usually contain a multi-agent system that can learn, an environment of some sort, and a communication system that allows the agents in the system to communicate about a set of predefined meanings. These meanings can be seen as ungrounded in the symbol grounding sense (Vogt, 2006). Vogt (2002, 2006) further argues that the symbol grounding problem can be transferred into a physical symbol grounding problem, where the construction of the relation between referent, meaning and form is very relevant, and that the symbols that the agents use must arise from the interaction between the agent and its environment. This is the point of view also adopted in this dissertation.

The models used in this dissertation are not ecological in a sense that they do not take into account the function of a language beyond naming observations. The 'mushroom world' experiments (e.g., Cangelosi and Parisi, 1998; Grim et al., 1999, 2004) often contain a simple simulated world, where the principal task of the agents is to survive. The agents might need to distinguish between edible and poisonous foods—often vi-

sualized as mushrooms, hence the name—or avoid being caught by predators. In these environments, the evaluation measure is the survival of the population, and language used is very simple. For example, in Cangelosi and Parisi (1998), only two signals are used. The hypothesis is that the language (signaling) skills help the agent population to spread the information, such as distinguishing predators, recognizing edible mushrooms, etc., faster, and thus enhance the rate of the survival in the population.

In the work presented in this dissertation, the focus is on modeling shared vocabulary emergence in a group of learners or agents, using language game model for communication.

3.4.1 Language games

In the context of this thesis, vocabulary acquisition has been modeled using *language games*. In this context, language games refer to a model setting in which there is a dialogue between a hearer and a speaker in a particular context to communicate about (Steels, 1996). The particulars of the game vary. These are, for example, what is in the context, whether the topic of the communication is explicitly defined, and whether feedback about the success of the communication is given, and in which form. In the computational language game models, the communicating agents are either simulated or robot agents. The shared context is an object or a group of objects in their presence. The purpose of the games is to learn a shared vocabulary.

The term *language game* was introduced by Wittgenstein (1963), who saw every occasion of the language use as a game: the meaning of words comes through their use. Vogt (2005) notes that it is realistic to assume that words and their meanings have co-developed in an embodied interaction of individuals with the real world.

In *associative learning*, words are associated with the meaning of referents that are simultaneously presented (Tomasello, 1999). In the language game formalism, this corresponds to an *observational* or *naming game*. There are many versions of the naming games, such as analogical naming games (Kaplan, 1998), multiple word naming games (Van Looveren, 1999), advertising games (Avesani and Agostini, 2003) and query-answering games (Agostini and Avesani, 2004). Naming games are used in conjunction with the Self-Organizing Map in Publications VII and VIII.

More complex types do exist as well. In a *guessing* game, the speaker and hearer are in the context of several objects the speaker can refer

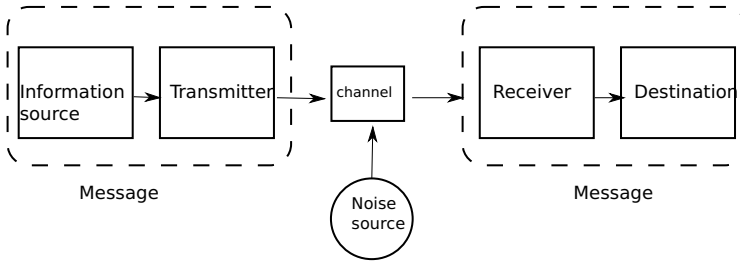


Figure 3.1. The standard model or the noisy channel model of communication

to. As the name suggests, the speaker does not inform the hearer of the topic of the game, but the hearer needs to guess which one the speaker refers to. After the game, the hearer receives corrective feedback, which makes learning in this game resemble *reinforcement learning* (Vogt and Coumans, 2003). (See also Section 4.3.) In real world, children sometimes receive corrective feedback on their language use.

Yet another type of a language game is based on the probability of co-occurrence of a word and a referent. It was first called the *selfish* game (Vogt, 2000), and later *cross-situational* game (Vogt, 2012). In such a game, words are observed in a context of many objects, and no direct association, nor corrective feedback is given. An agent learns to associate a given word to a given reference based on the probability of the occurrence of the word and the reference. These types of games only consider vocabulary acquisition in a community. In a more general, game theoretic setting, the pragmatics can also be considered, but those go beyond the scope of this work. For example, Benz et al. (2006) can be used as a starting point for that domain.

3.4.2 The standard model for communication

The standard model of communication (Shannon, 1948) is shown in Figure 3.1. The model of a noisy channel contains several parts. A sender sends a message using a coder via a channel, where noise may be added to the message. The receiver receives the message via a decoder. This model of communication is used in engineering as a standard model of information theory, but also in social sciences as a more general framework for describing communication between individuals or groups. This model does not take into account the receiver, or the semantic contents of the model, which are outlined in the problem of three levels of communication, or the Shannon-Weaver model (Weaver, 1949):

Level A: How accurately can the symbols of communication be transmitted? (The technical problem.)

Level B: How precisely do the transmitted symbols convey the desired meaning? (The semantic problem.)

Level C: How effectively does the received meaning affect conduct in the desired way? (The effectiveness problem.)

The semantic problems are concerned with the identity or satisfactorily close approximation in the interpretation of the meaning by the receiver, as compared to the intended meaning of the sender.

4. Methods

The two previous chapters discussed the linguistic, cognitive science and modeling perspectives of the topic at a theoretical level. This chapter introduces a collection of computational tools with the intent of providing some methodological background and mathematical notation to understand the experiments in the following chapters. First, the basic principles of random variables, probability distributions, information theory, and machine learning are introduced to the extent they are relevant for this dissertation. Vector space models are a central tool used to represent the meaning of words using textual data. In the previous chapter, we briefly discussed them from the modeling point of view. In this chapter, their construction is discussed in detail. Most machine learning methods used in this dissertation are unsupervised, and they are introduced at the end of this chapter.

4.1 Random variables and distributions

Statistical methods are at the core of statistical natural language processing, in which word frequencies and distributions are used. In this section, a variety of concepts from probability theory will be covered. For a more thorough account, the reader is referred to any standard textbook on probability theory, such as Papoulis (1991).

The value of a *random variable*, X changes due to chance. The observation of a random variable X is denoted with x. The probability or density function of a random variable is written as $p(X = x)$, usually written in the shortened form $p(x)$.

Distributions can be defined for a single variable (univariate) or multiple variables (multivariate) case. A *random vector*, is a vector $[X_1, \ldots, X_k]$ where the X_k are random variables. Vector representations are marked in

boldface throughout this thesis. Numerical data are seen as drawn from some assumed probability distribution, where the shape and type of the distribution depends on the generating process. The distributions of random variables can be divided into discrete and continuous, depending on the types of values the random variable can have. The distributions are often defined by the probability mass function of the distribution in the discrete case, or the probability density function in the case of continuous distributions. For discrete distributions, $\sum p(x) = 1$ and for continuous distributions $\int p(x)dx = 1$.

4.1.1 Discrete distributions

The Bernoulli distribution is the distribution of a single random variable with two values, $X \in \{0, 1\}$, with $p(X = 1) = \pi$.

Categorical distribution is an extension of the Bernoulli distribution, describing the outcome of a random event that can take on one of k possible outcomes specifying the probability of each outcome separately:

$$p(X = x_i) = \pi_i, \tag{4.1}$$

where $\boldsymbol{\pi} = [\pi_1, \dots \pi_k]$, $\sum_i \pi_i = 1$ is the vector of parameters.

The binomial distribution shows the distribution of a series of independent Bernoulli trials. The probability of having the event $X = 1$ occurring in k of the n trials is given by

$$p(k|n, \pi) = \binom{n}{k} \pi^k (1 - \pi)^{n-k}. \tag{4.2}$$

For example in text data, sentences are not actually independent of each other, but after a while the dependence effect disappears, and hence the binomial distribution can be used to model, for example, finding examples of a certain word in sentences (Manning and Schütze, 1999).

Multinomial distribution is a generalization of the binomial distribution with n trials for a case in which X_i is the number of occurrences of the event i.

$$p(X_1 = x_1, X_2 = x_2, \dots, X_k = x_k|\boldsymbol{\pi}, n) = \binom{n}{x_1 x_2 \dots x_i} \prod_{i=1}^{k} \pi_i^{x_i}. \tag{4.3}$$

In NLP applications, the bag of words model of a document, which loses the order information, can be modeled with a multinomial distribution.

4.1.2 Continuous distributions

The normal or Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ is defined by its mean μ and variance σ^2. Its probability density function is given as

$$\mathcal{N}(X = x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}. \tag{4.4}$$

Gaussian distribution has several important properties. First, Gaussian distributions can be fully described by only the first and second order statistics, that is, the mean and variance. Second, linear transformations of variates with Gaussian distribution are also Gaussian. The *central limit theorem* is also relevant when discussing Gaussian distributions: the sum of a set of random variables has a distribution that becomes increasingly Gaussian when the number of terms in the sum increases.

Dirichlet distribution is a multivariate distribution with k variables with parameters $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_k]$ in the exponential family. Assuming the probabilities of Eq. (4.1), the probability density function of the Dirichlet distribution is

$$p(\pi_1, \ldots, \pi_k|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_k)} \prod_{i=1}^{k} \pi_i^{\alpha_i - 1}, \tag{4.5}$$

where $\Gamma(x) = \int_0^\infty u^{x-1} e^{-u} du$ is the Gamma function. Dirichlet distribution is conjugate (see Section 4.1.5) to the parameters of multinomial distribution. This makes it very useful as a prior distribution, as many properties of natural languages can be modeled with the multinomial distribution.

4.1.3 Moments

The moments that describe the 'shape' of a probability distribution can be defined as follows. The j^{th} moment α_j of X is defined by the expectation (Hyvärinen et al., 2001)

$$\alpha_j = E\{X^j\} = \int_{-\infty}^{\infty} x^j p(x) dx, \text{ j = 1, 2, } \ldots \tag{4.6}$$

and j^{th} central moment is

$$\mu_j = E\{(X - \alpha_1)^j\} = \int_{-\infty}^{\infty} (x - \alpha_1)^j p(x) dx, \text{ j = 1, 2, } \ldots. \tag{4.7}$$

Thus the mean of X equals the first moment, α_1. The second central moment, μ_2, is the variance.

Nongaussianity of a distribution

In Independent Component Analysis that will be described later, nongaussianity is an important concept, as Independent Component Analysis can only be defined for nongaussian sources—or more specifically, only one source can be Gaussian.

The third central moment, *skewness*, is a useful measure of the asymmetry of the probability density function.

$$\mu_3 = E\{(X - \alpha_1)^3\}. \tag{4.8}$$

It is zero for symmetric densities such as Gaussian. Kurtosis for a zero-mean case can be defined as (Hyvärinen et al., 2001):

$$\text{kurt}(X) = E\{X^4\} - 3[E\{X^2\}]^2 = \alpha_4 - 3\alpha_2^2. \tag{4.9}$$

and it is zero for Gaussian, but nonzero for most nongaussian random variables. Random variables with positive kurtosis are called subgaussian and have a higher peak and fatter tails of the distribution, whereas random variables with negative kurtosis are called subgaussian and characterized by a rounder peak and thinner tails.

4.1.4 Joint distributions, uncorrelatedness and independence

Two random variables, X and Y, can have a joint distribution $p(X, Y)$. This can be calculated as the product of the probability of one variable and the conditional probability of the other given the first:

$$p(X, Y) = p(X)p(Y|X) = p(Y)p(X|Y). \tag{4.10}$$

Two random variables X and Y are uncorrelated, if their covariance c_{XY} is zero

$$c_{XY} = E\{(X - \mu_X)(Y - \mu_Y)\} = 0, \tag{4.11}$$

where $\mu_X = E\{X\}$ and $\mu_Y = E\{Y\}$.

Independence is a stronger property than uncorrelatedness. Two random variables, X and Y are statistically independent if and only if their joint density $p(X, Y)$ is the product of their marginal densities $p(X)$ and $p(Y)$:

$$p(X, Y) = p(X)p(Y). \tag{4.12}$$

In other words, the value of a random variable X does not give any infor-

mation of a random variable Y and vice versa, if they are independent.

4.1.5 Bayes' theorem

There is a relationship between conditional probabilities of X and Y. From Equation 4.10, it holds that

$$p(X|Y) = \frac{p(X)p(Y|X)}{p(Y)}. \tag{4.13}$$

Here $p(X)$ is the *prior* and $p(X|Y)$ the *posterior* probability after observing Y. Bayesian statistics sees the probabilities as a measure of uncertainty. Bayes' rule can then be used to update the probability estimate for a hypothesis in the light of new evidence. The Bayesian statistics have many applications in Natural Language Processing, among others finding topics (Griffiths et al., 2007), or part of speech tagging (Goldwater and Griffiths, 2007).

A posterior distribution and a prior distribution are called conjugate distributions if they come from the same distribution family and thus have the same functional form (Bishop, 2006). The prior is called the *conjugate prior* of a likelihood function. This means that when updating the estimation based on observations, the posterior distribution can be solved analytically if the prior is chosen wisely. Gaussian distributions are self-conjugate: if the likelihood function is Gaussian, selecting a Gaussian prior also ensures a Gaussian posterior. The likelihood and the prior do not need to be from the same family. For example, a Dirichlet prior is a conjugate of a multinomial likelihood. This pair is used in probabilistic topic modeling in Section 4.6.6.

4.2 Concepts in information theory

Information theory was developed to find a theoretical maximum amount of information that can be transmitted over a noisy communication channel, and to find a code that is suitable for a data set with certain statistical properties (Shannon, 1948; Hyvärinen et al., 2001).

Entropy is the average uncertainty of a single random variable. The more random or unpredictable the variable is, the larger its entropy. En-

tropy is usually measured in bits, i.e. with the logarithm of base 2.

$$H(X) = -\sum_{x} p(x) \log_2 p(x). \tag{4.14}$$

Mutual information measures the information that the members of a set of random variables have on other random variables in the set, or the dependence between variables in that set.

$$I(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} H(X_i) - H(X_1, \ldots, X_n), \tag{4.15}$$

where $H(X)$ is the entropy of X.

4.2.1 Kolmogorov complexity

Kolmogorov complexity (Kolmogorov, 1998; Li and Vitanyi, 1997) of a sequence is the length (in bits) of the shortest computer program that prints the sequence and then halts. For predictable sequences the algorithm is shorter than for random ones, and thus the Kolmogorov complexity is lower. Compared to (Shannon's) information theory, Kolmogorov complexity considers the information of individual objects. Kolmogorov complexity is incomputable, but approximations exist (Grünwald and Vitanyi, 2003).

4.2.2 Data compression

The purpose of data compression is to encode the information in given data using a shorter form than originally, by exploiting regularities in the data. A good data compression system can be used to approximate Kolmogorov complexity (Juola, 2008). Different compression algorithms have been devised. They can be divided to lossless and lossy algorithms. In lossless compression, the original data can be returned, whereas in lossy compression, some of the data is irretrievably lost. Lossless compression algorithms are, for example, Huffman coding (Huffman, 1952), Lempel-Ziv-Welch algorithms (Ziv and Lempel, 1978; Welch, 1984), and the Burrows-Wheeler algorithm (Burrows and Wheeler, 1994). In this thesis, the bzip2 compression based on Burrows-Wheeler algorithm is used. As a detailed account on compression algorithms, for example, Moffat and Turpin (2002) can be used.

4.2.3 Minimum description length

Minimum Description Length (MDL) (Rissanen, 1978) is a general principle for doing inductive inference (Grünwald, 2005). It uses the insight that 'learning may be viewed as data compression'. In an ideal case, MDL would be equivalent to Kolmogorov complexity. In practice, the principle tells us that for hypotheses H and data set D we should find the hypothesis or a combination of hypotheses in H that compresses D the most. If $L(H)$ is the length of the hypothesis in bits, and $L(D|H)$ is the length of the description of the data when encoded with the hypothesis, the 'crude' MDL searches to minimize the sum $L(H) + L(D|H)$. The Morfessor method that will be introduced in Section 4.6.1 uses the MDL principle.

4.3 Machine learning

The basic idea behind machine learning is that we expect that there is a process that explains the observed data (Alpaydin, 2004). We do not know the details of the process, but we know or hope that it is not completely random: there are regularities we can hope to find.

Machine learning paradigms differ in their use of corrective feedback. By definition, in a *supervised* learning, a system, for example, a classifier or a neural network model, learns a mapping $\hat{y} = f(x)$ from x to y given a training set of pairs (x_i, y_i). In other words, the system aims to learn to map an input to a correct output, after which the system can predict the output y for a new input x. An example of an NLP application is a system which learns to classify email into valid email and spam based on examples of each class, where the emails are represented as some kind of feature vectors. Providing the pre-classified data is often expensive and time consuming, which limits the applicability of supervised learning.

In *unsupervised* learning, correct labels are not used in learning. Instead, the task is to find interesting properties from a set of observations $\mathbf{X}$. The unsupervised learning task may be seen as density estimation of the underlying probability density that produces $\mathbf{X}$ or, in weaker form, *clustering*. For a thorough review on unsupervised learning approaches, see Oja (2002). In this dissertation, the methods based on unsupervised learning are largely used, but vocabulary learning in the agent simulation experiments contains a supervised component.

In *semi-supervised* learning (Chapelle and Zien, 2006; Zhu, 2008), only part of the labels is used in training. The use of unlabeled data can improve the classifier if the unlabeled samples allow the classifier to model the input distributions better. Often there is also a lack of labeled data, and then even a small amount of labeled data can make the system perform better than when only using unlabeled data in unsupervised way. An example of such a system is a semi-supervised version of morphology analyzer, where only a small number of linguistic gold standard labels are available (Kohonen et al., 2010).

Reinforcement learning (Sutton and Barto, 1998) approach is quite different. In it, the whole problem of interaction with environment is considered. The learner functions in an environment that can change, and the learner has a goal to reach. The feedback available for the learner is a reward signal, and the learner must learn to choose those actions that maximize the total reward. Thus, learning must proceed with trial and error, and the learner must balance between using the actions it already knows and exploring for possible better choices.

4.4 Distributional similarity: word vector space models

The modeling principles of the word space models were introduced in Section 3.2.2. They are based on the general hypothesis that statistical patterns of human word usage can be used to find out what is meant by a word. In the following, the components needed to build such a model are described in more detail.

Schütze (1993) introduced a way to obtain lexical co-occurrence statistics using large-scale linear regression. He used letter-fourgrams and showed that words that are semantically similar tend to be close together in the vector space, whereas unrelated words are distant. In a related work, Ritter and Kohonen (1989) used simple English sentences to show that similar parts of speech organize close to each other in a SOM based on context information only. In current works, words are often used, with more or less preprocessing. For example, lowercasing all words, or stemming or lemmatisation can be performed.

The model construction has several steps. First, the text data is preprocessed and potentially a feature selection can be applied. The context word frequencies are calculated, and raw frequency counts are transformed by weighting. A dimensionality reduction can be applied to smooth

the space. Finally, the similarities between word vectors are calculated (Turney and Pantel, 2010).

The phases can also be defined mathematically. Lowe (2001) defines the word vector space model as a quadruple $\langle A, B, S, M \rangle$, where B is the set of features used to represent the words. B is usually the context word vocabulary. S is the similarity measure calculated between pairs of word vectors and A is the weighting function used to change the raw co-occurrence counts into association weights. M is then a transformation of the whole vector space, for example, by using dimension reduction. In the following, each of these steps are considered in more detail.

4.4.1 Data and pre-processing

Symbolic data needs to be transformed into numerical form. In the case of vectorial representations, the properties used to describe the instances are called *features*. A vector is a collection of these features. A data matrix can then be obtained by combining the vectors as the rows of a matrix.

The first vector space models utilized document-term or document-word matrices. The rows of the matrix represented the documents D in the collection. Document representations are often used in information retrieval tasks. Each document d_j is represented by the values of the column features f_i, which are the counts of the terms that appear in the document. The feature value is simply the number of times the i-th term w_i appears in document d_j. This kind of representation is called the *bag-of-words* model (Salton et al., 1975), as the order of the words is discarded. The order of words and encoding in a sentence or document carries of course information, but the model works surprisingly well. Similarly, words can be represented as vectors: A word in a document context can be represented by either transposing the document-term matrix and representing the words by the information of which documents they appear in, or by building separately a word-word matrix, in which word co-occurrence counts are calculated.

The word frequencies need to be high enough to obtain reliable representations. Thanks to the rise of the availability of textual documents in electronic form, large corpora are more easily accessible. In this dissertation, two different corpora have been used. The Europarl corpus (Koehn, 2005), which contains the Proceedings of the European Parliament sentence-aligned in many European languages, has been used for the bilingual setting in Publication II and a corpus collected from the En-

Table 4.1. Construction of a co-occurrence count matrix

(a) Illustration of the sliding window of size 3 over a sample sentence 'a quick brown fox jumped over a lazy dog'.

1	a	quick	brown					
2		quick	brown	fox				
3			brown	fox	jumped			
4				fox	jumped	over		
5					jumped	over	a	
6						over	a	lazy
7							a	lazy	dog

(b) The co-occurrence matrix from the sample sentence.

	a	brown	dog	fox	jumped	lazy	over	quick
a	0	0	0	0	0	1	1	0
brown	0	0	0	1	0	0	0	1
dog	0	0	0	0	0	0	0	0
fox	0	1	0	0	1	0	0	0
jumped	0	0	0	1	1	0	0	0
lazy	1	0	1	0	0	0	0	0
over	1	0	0	0	1	0	0	0
quick	1	1	0	0	0	0	0	0

glish Wikipedia has been used in Publications III–V.

Some simple pre-processing steps are also needed. The text data is usually split into tokens. Often tokenization is carried out by cutting the text stream at white spaces or at non-alphanumeric characters for Western languages, where word boundaries are more clearly marked. The text can also be lemmatized, i.e. returned to base form, or stemmed. In the Publications of this dissertation, a simple tokenization based on white spaces between words and punctuation marks has been used.

4.4.2 Context

The choice of a *context* is central when building a word vector space model. The context is defined as the surrounding words that are taken into account when calculating the co-occurrence counts for a target word. Typical choices for context are whole documents, sentences, or short windows around the target word. Different contexts give different representations with different information contained in them. Clark (forthcoming) points out that when the context is as large as a sentence, topical similarity, for example, relating car and gasoline, is found. Sahlgren (2006) points out that small contexts seem to give rise to more paradigmatic relations between words, whereas large contexts create representations with more syntagmatic relations (Section 2.2.1).

The co-occurrence counts are often constructed using the window method. In this method, a sliding window of fixed size is used. The process of constructing a co-occurrence count matrix for a sample sentence is shown in Table 4.1. Table 4.1a shows the progress of the sliding window of three words used for a sample sentence 'a quick brown fox jumped over a lazy dog'. A symmetrical window of size three means that the occurrences of one word to the left and one to the right around a target word will be counted including only full windows. In literature, this window size is often referred as $1 + 1$ (e.g., Bullinaria and Levy, 2007). The resulting co-occurrence count matrix is shown in Table 4.1b.

The representation can also contain multiple languages. Publication II is an example, where the authors used a sentence context in a bilingual (Finnish-English) case. In that case, target words in each language were simply represented by a sentence context in the two languages.

More sophisticated methods besides a fixed window can also be used. Sahlgren (2006) and Bullinaria and Levy (2007) also experiment with weighted window, where words further away from the target words have less effect. Bullinaria and Levy (2007) conclude that the effect of the weighting is very small compared to an unweighted window of a smaller size. One can also specify the positions of words in the window, for example, treating left and right contexts separately (Bullinaria and Levy, 2007). A windowless approach has also been introduced (Washtell, 2009; Washtell and Markert, 2009). Turney and Pantel (2010) give a good review on different uses of context for term-document, term-term and pair-pattern matrices.

4.4.3 Feature selection and weighting

The purpose of both feature selection and weighting is to provide statistically reliable representations. In a vector representation, the most frequent words appear in almost any context, which reduces their usefulness in a semantic representation. Based on information theoretic principles of a surprising element carrying more information (Shannon, 1948), we want to be able to give more weight to surprising elements than unsurprising elements (Turney and Pantel, 2010). Moreover, words that are not frequent enough are not trustworthy features and add noise to the representation.

If the data is labeled, supervised feature selection can be carried out. The purpose is to retain those features that are most informative. Two

Table 4.2. Local and global weighting schemes for word w_i in document d_j, with $j \in (1, \ldots, N_d)$, $i \in (1, \ldots, N_w)$. The total number of word occurrences in different contexts is $\sum_{i=1}^{N_w} \sum_{j=1}^{N_d} f_j(w_i) = M$

Local weighting	
term frequency, tf	$f_j(w_i)$
logarithmic tf, log-tf	$\log(1 + f_j(w_i))$
Global weighting	
collection frequency, $cf(w_i)$	$\sum_{j=1}^{N_d} f_j(w_i)$
document frequency, $df(w_i)$	$\sum I(f_j(w_i) > 0)$
inverse document frequency, idf	$\frac{N_d}{df}$
logarithmic idf	$\log(\frac{N_d}{df})$
entropy	$\sum_{j=1}^{N_d} \frac{p_{ij} \log(p_{ij})}{\log(N_d)}$, where $p_{ij} = \frac{f_j(w_i)}{cf(w_i)}$
pointwise mutual information, PMI	$\log\left(\frac{p(w_i, d_j)}{p(w_i)p(d_j)}\right)$
positive PMI, PPMI	$\begin{cases} pmi_{ij}, & \text{if } pmi_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$
Combined	
tf.idf	$log(1 + f_j(w_i) \log\left(\frac{N_d}{df}\right)$

approaches, either *forward* or *backward* can be used. In forward selection, features are added one by one, by always adding a feature based on a certain selection criterion. Similarly, in backward selection the features are removed one by one, removing the features based on a selection criterion (Alpaydin, 2004). Different criteria, such as correlation, mutual information or class separability exist for that purpose (Guyon and Elisseeff, 2003).

When building vector space models, unsupervised feature selection heuristics are often used. The most frequent words, which are often function words such as 'a', 'an', or 'that' and co-occur with most of the words, are often directly culled out of the representation by using so called *stop-word* lists, as the purpose is to obtain a representation of meaning of words. To eliminate the noise caused by the low frequency words, a cutoff threshold is often specified, and words with a frequency below that threshold are cut off. This also has the effect of reducing the dimensionality of the representation, as the number of low frequency words is high, according to Zipf's law. Features can be also selected on more distinct criteria: for example, using a certain syntactical relation or part of speech (Clark, forthcoming)

Instead of or in addition to such lists, different weighting schemes are often used to dampen the effect of the most frequent words. Local weighting schemes are calculated within a document or context, whereas global weighting schemes are calculated over the whole collection. Typical weighting schemes used for vector space models are listed in Table 4.2.

Let us define the frequency or the number of occurrences of a word w_i in a document d_j as $f_j(w_i)$. This is the term frequency (tf). The document frequency $df(w_i) = \sum I(f_j(w_i) > 0)$ is a global weight, and it is calculated as the number of documents in the collection the word w_i occurs in. The number of different word types in the vocabulary is N_w, and the total number of word occurrences in the co-occurrence count matrix is M. Inverse document frequency is the total number of documents N_d, divided by the document frequency: $\frac{N_d}{df}$. Collection frequency $cf(w_i)$ is the total number of occurrences of the word w_i appearing in the whole collection. Term frequency and inverse document frequency are often combined as a joint weight that works both in local and global scale, taking into account the frequency of a term in one document and in the whole collection. Term Frequency - Inverse Document Frequency (tf.idf) is a commonly used weighting scheme for term-document matrices (Jones, 1972; Salton and Buckley, 1988). It combines the local weighting with a global weighting scheme which smooths the effect of frequent words in the corpus. Manning and Schütze (1999) remark that tf is often dampened with a logarithm. Similarly, inverse document frequency can be either linear, or logarithmic.

Entropy weighting takes into account the distribution of the occurrences of the terms. It assigns a minimum weight to terms which are evenly distributed in the documents and a maximum weight to those terms that are concentrated on few documents.

Pointwise Mutual Information (PMI) (Church and Hanks, 1989; Turney, 2001) can be also used as a weighting scheme. It has been demonstrated to work well for both term-document (Pantel and Lin, 2002a) and word-context matrices (Pantel and Lin, 2002b). In the pointwise mutual information definition, $p(w_i, d_j)$ is the probability that word w_i occurs in context or document d_j. $p(w_i)$ is the estimated probability of the word w_i, $p(d_j)$ is the estimated probability of the context or document d_j. If the word w_i and the context d_j are independent, $p(w_i, d_j) = p(w_i)p(d_j)$. If $p(w_i, d_j) >> p(w_i)p(d_j)$, (or $p(w_i, d_j) << p(w_i)p(d_j)$), there is some (semantic) relation between the word w_i and the context d_j (Turney and Pantel, 2010). If word w_i and context d_j are completely unrelated, PMI may give negative values. The Positive Pointwise Mutual Information (PPMI) weighting solves this problem by only retaining the non-negative values of PMI (Niwa and Nitta, 1994). It also performs better with word-context matrices compared to many different weighting schemes in the semantic

evaluation tests (Bullinaria and Levy, 2007). Bullinaria and Levy (2012) concluded that removal of stop words or the most frequent words does not improve the evaluation results. This might be due to the fact that the weighting schemes often used already lower the weight of the most frequent words which are often in the stop word list. As a consequence, removing these words does not improve results any further.

4.4.4 Feature extraction and dimensionality reduction

Word vector space models often have a very high dimensionality. This is why several feature extraction or dimensionality reduction methods have been proposed. The two main approaches to reduce the high dimensionality are *feature selection* described in the previous section and feature extraction, which refers finding a new set of features that are combinations of original dimensions (Alpaydin, 2004).

The most common of them is the Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997) also known as Latent Semantic Indexing (LSI), based on Singular Value Decomposition (SVD). This method will be described in more detail in the following sections of this chapter. It was first used for document-term-matrices, in which it finds the second-order relations between words: i.e. relation on not only which words occur together but which words share similar contexts, similarly to what is achieved when compiling a word-word co-occurrence count matrix. For word-word matrices, LSA can also be used for reducing dimensionality of the representation. The SVD, Principal Component Analysis (PCA), Independent Component Analysis (ICA) and the probabilistic Latent Dirichlet Allocation (LDA) will be introduced in the following sections of this work.

Recently, neural network models trained with context information have been introduced. In a sense, they also produce a low-dimensional representation from high-dimensional data, even though the process is nonlinear. In that field, the resulting low-dimensional vector representations are called 'embeddings'. The results in different NLP tasks have been very good, but the models suffer from a long training time (Collobert, 2011; Collobert et al., 2011).

4.4.5 Measuring similarity

The similarity of the word vectors is measured by calculating the distance between them. Some commonly used measures are given in Table 4.3. The

Table 4.3. Some common distance measures for vector space models

Non-probabilistic	
Cosine	$d(\mathbf{x},\mathbf{y}) = 1 - \frac{\sum x_i y_i}{\sqrt{\sum x_i^2}\sqrt{\sum y_i^2}}$
Euclidean	$d(\mathbf{x},\mathbf{y}) = \sqrt{\sum (x_i - y_i)^2}$
City-block	$d(\mathbf{x},\mathbf{y}) = \sum \lvert x_i - y_i \rvert$
Probabilistic	
Kullback-Leibler	$d(\mathbf{x},\mathbf{y}) = \sum p(x_i) log(\frac{p(x_i)}{p(y_i)})$

most commonly used in VSM models is the Cosine (Turney and Pantel, 2010), which measures the angle between the vectors. It has the benefit of normalizing the length of the vectors as well. In VSM applications, the values of cosine will be limited to positive range between $[0, 1]$, defining cosine distance as $1 - \cos(\mathbf{x}, \mathbf{y})$.

Geometric distances, such as the Euclidean distance or the City-block distance can also be used. The use of Euclidean distance may be problematic if the vector lengths of the two vectors are very different. For normalized vectors such that $\sum x_i^2 = \sum y_i^2 = 1$, Euclidean distance and Cosine distance give obviously the same ranking, because then $\sqrt{\sum (x_i - y_i)^2}$ = $\sqrt{\sum x_i^2 + \sum y_i^2 - 2\sum x_i y_i}$ = $\sqrt{2(1 - \sum x_i y_i)}$ = $\sqrt{2(1 - \cos(\mathbf{x}, \mathbf{y}))}$. City-block distance or the L_1 norm measures the absolute difference of the coordinate values, and is also sometimes used with vector space models. For binary vectors, also Dice and Jaccard coefficients can be calculated (Manning and Schütze, 1999). Word vectors are made of counts and can be interpreted as multinomial probabilities, and probabilistic metrics such as Kullback-Leibler divergence can also be used. Bullinaria and Levy (2007) compared several of these methods in different tasks, and found out that the cosine distance paired with a positive pointwise mutual information weighting gave the best overall results. Turney and Pantel (2010) suggest that determining the most appropriate measure is dependent on the task, the sparsity of the statistics, the frequency distribution of the elements, and the feature extraction method used.

4.5 Evaluation

To know how a computational model performs, it must be evaluated. In NLP applications, the performance is usually compared to some linguistic resources. The evaluation can be direct (intrinsic), or indirect (extrinsic) (Sahlgren, 2006; Suominen et al., 2008). In direct evaluation, the perfor-

	Class positive	Class negative
Test positive	true positive (tp)	false positive (fp)
Test negative	false negative (fn)	true negative (tn)

Table 4.4. The confusion matrix

mance is compared to a test set directly. In indirect evaluation, the performance of the component is evaluated in an application, for example, if no direct test sets are available.

In classification tasks, two core concepts, Precision (Pr) and Recall (Re) are defined. There are four categories of outcomes in a classification task. A true positive (tp) and true negative (tn) are the correct classification outcomes, whereas false positive (fp) and false negative (fn) are the classification errors, see Table 4.4. Thus, Precision is defined as the number of true positives divided by all instances labeled positive.

$$\mathrm{Pr} = \frac{tp}{tp + fp} \tag{4.16}$$

Recall is the number of true positives divided by true positives and false negatives, i.e. all the instances that have a positive label.

$$\mathrm{Re} = \frac{tp}{tp + fn} \tag{4.17}$$

Precision and Recall can be combined into an *F-score*, which is the harmonic mean of the two.

$$F = 2 \cdot \frac{\mathrm{Pr} \cdot \mathrm{Re}}{\mathrm{Pr} + \mathrm{Re}} \tag{4.18}$$

The same concepts are used in information retrieval, and defined slightly differently. Let R be the set of retrieved documents we have obtained, with $|R| = tp + fp$, and Q be the set of documents that are deemed relevant for that given query, with $|Q| = tp + fn$. Then Precision is the number of documents that are both relevant and retrieved divided by the number of retrieved documents.

$$\mathrm{Pr} = \frac{|R \cap Q|}{|R|} \tag{4.19}$$

Recall is the number of relevant and retrieved documents divided by the number of relevant documents.

$$\mathrm{Re} = \frac{|R \cap Q|}{|Q|} \tag{4.20}$$

4.6 Unsupervised learning methods used in this thesis

4.6.1 The Morfessor method

The Morfessor is a method for unsupervised induction of simple morphology from corpora. It is a data-driven method that produces a segmentation of corpus into smaller elements. These elements often resemble linguistic morphs (Lagus et al., 2005), and are therefore called morphs in this context as well. The Morfessor exists in different variations. The one used in Publication I is the Morfessor Baseline. The method is inspired by the MDL principle described in Section 4.2.3, and it uses Maximum a Posteriori (MAP) framework (Creutz and Lagus, 2007).

Maximum a posteriori estimation
MAP models try to find the best model jointly considering model accuracy and model complexity (Creutz and Lagus, 2007). The MAP estimate for the model to maximize is

$$\arg\max_{\mathcal{M}} P(\mathcal{M}|\text{data}) = \arg\max_{\mathcal{M}} P(\text{data}|\mathcal{M}) \cdot P(\mathcal{M}), \tag{4.21}$$

where $P(\mathcal{M})$ is the probability of the model. In the case of the Morfessor Baseline, $P(\mathcal{M}) = P(\text{lexicon, grammar})$ is the probability for the model of language, which is a joint probability of the induced lexicon and grammar. Virpioja (2012, Section 6.4) offers a thorough introduction to the mathematical formulation behind the Morfessor method family.

In the Baseline model, only the morph frequency is taken into account, and there is only one morph category, as opposed to advanced methods, where types of morphs are defined as stem, prefix and suffix. In the Baseline, the words are split into frequently occurring strings (Creutz and Lagus, 2005). The model first considers each word as a whole and adds it to the morph lexicon. Every possible split into two sub-strings is evaluated, and the split that has the highest probability is selected. The case of no split is also possible. In the case of a split, the splitting of the two parts continues recursively until no more gains are obtained. The words in the corpus are reprocessed until the overall probability converges. Creutz and Lagus (2007) note that as many of the word stems occur in isolation, the algorithm finds suffixes and prefixes by splitting of known stems from longer words, and the newly discovered morphs can then be used to split other words. Therefore, the overall task is a complex search problem.

4.6.2 Singular Value Decomposition and Latent Semantic Analysis

Singular value decomposition (SVD) is a matrix decomposition method for any rectangular matrix. In the context of vector space models for documents, Latent Semantic Analysis (LSA) uses the SVD technique (Landauer and Dumais, 1997) on term-document matrices, but SVD can be also performed on term-term matrices.

The singular value decomposition is defined as

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T, \tag{4.22}$$

where $\mathbf{D}$ is a diagonal matrix of the singular values, which are real and non-negative, and $\mathbf{U}$ and $\mathbf{V}$ are orthonormal column matrices, i.e., $\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I}$.

In practice, only d first singular values are calculated, i.e. a truncated SVD is used. Truncated SVD approximates the matrix $\mathbf{X}_{C\times N}$

$$\mathbf{X}_{C\times N} = \mathbf{U}_{C\times d}\mathbf{D}_{d\times d}\mathbf{V}^T_{N\times d}, \tag{4.23}$$

Thus, $\mathbf{D}_{d\times d}$ is a diagonal matrix of the d largest eigenvalues of $\mathbf{X}^T\mathbf{X}$ (or $\mathbf{X}\mathbf{X}^T$), and $\mathbf{U}_{C\times d}$ has the d corresponding eigenvectors of $\mathbf{X}\mathbf{X}^T$, and $\mathbf{V}_{N\times d}$ has the d corresponding eigenvectors of $\mathbf{X}^T\mathbf{X}$.

In the reduced space, the $\mathbf{V}_{N\times d}$ gives a d-dimensional representation for the target words. This representation is said to be *latent*, as it represents the words (or documents) in a new space of *latent* features.

4.6.3 Principal Component Analysis

Principal Component Analysis (PCA) is a feature extraction method originally proposed by Pearson (1901) and Hotelling (1933), and widely used for dimensionality reduction, data compression, feature extraction and data visualization (Bishop, 2006). It does not select the best features of the original dimensions like feature *selection*. Instead, it is a projection method that searches for a mapping from the original dimensions to a new space of smaller dimensionality with a minimum loss of information. Principal Component Analysis uses the criterion of maximizing the variance. Thus, the projection of the data along the first principal component $\mathbf{w}_1$ has the largest variance (Alpaydin, 2004). The second principal component should also maximize variance, and be orthogonal to $\mathbf{w}_1$, because the

projections along $\mathbf{w_1}$ and $\mathbf{w_2}$ should be uncorrelated. PCA is calculated by first normalizing it with regards to the first order statistics, i.e. centered by subtracting its mean. PCA is not often applied to co-occurrence data due to the centering. The co-occurrence matrices are usually sparse, but centering loses this property, making it computationally complex.

4.6.4 Independent Component Analysis

Independent Component Analysis is an unsupervised learning method used in this thesis. It is applied to a blind source separation task, where one tries to find the independent sources from mixed observed signals (Comon, 1994; Hyvärinen et al., 2001).

The basic ICA equation

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{4.24}$$

represents each observed random variable $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ as a weighted sum of independent random variables $\mathbf{s} = (s_1, \ldots, s_k, \ldots, s_n)^T$. $\mathbf{A}$ is the mixing matrix that contains the weights. In ICA, both the $\mathbf{s}$ and $\mathbf{A}$ are estimated, which means that there are ambiguities which cannot be resolved: 1) the variance of the independent component, 2) the sign of the component and 3) the order of the components. The variance of each independent component is usually set to unit variance. The model has some further requirements: 1) The independent components are assumed to be statistically independent. The requirement for independence is stronger than in PCA, where only uncorrelatedness is expected. 2) The problem is well-defined if and only if at most one of the components of $\mathbf{s}$ is Gaussian (Hyvärinen et al., 2001).

ICA can be estimated with different approaches. It can be estimated by using a heuristic to maximize the nongaussianity: finding independent components one by one with each local maximum giving one independent component. Here a method of estimation using negentropy is described. A Gaussian variable has the largest entropy among all random variables of equal variance. Negentropy can thus be defined to be the difference between the entropy of a Gaussian random variable y_{gauss} with the same correlation and covariance as y:

$$J(y) = H(y_{gauss}) - H(y), \tag{4.25}$$

where $H(y) = -\int p(y) \log p(y) \, \mathrm{d}y$. The negentropy estimation is computationally difficult as an estimate for the probability density function would

be needed, but negentropy can be approximated by

$$J(y) \propto [E\{G(y)\} - E\{G(\nu)\}]^2, \tag{4.26}$$

where G is a non-quadratic function such as $\tanh$ and ν is a Gaussian variable of zero mean and unit variance and y is assumed to have zero mean and unit variance as well.

ICA can be also defined more rigorously through information theoretic principles, without making many assumptions about the data itself. Mutual information measures the dependence between random variables and as such, it can be used as a criterion for finding the ICA representation (Hyvärinen et al., 2001).

Practical considerations

Independent Component Analysis is not a dimensionality reduction method, and usually the data dimensionality is reduced in a pre-processing step with PCA.

ICA problem is simpler if the observed mixture vectors are first whitened. A zero-mean random vector $\mathbf{z}$ is white, if its elements are uncorrelated and have unit variances (Hyvärinen et al., 2001). In other words, the covariance matrix of $\mathbf{z}$ is the identity matrix:

$$E\{\mathbf{z}\mathbf{z}^T\} = \mathbf{I}. \tag{4.27}$$

In practice, a vector $\mathbf{x}$ can be whitened with a whitening matrix $\mathbf{V}$,

$$\mathbf{z} = \mathbf{V}\mathbf{x} = \mathbf{D}^{-1/2}\mathbf{E}^T, \tag{4.28}$$

where the columns of $\mathbf{E} = (\mathbf{e}_1 \dots \mathbf{e}_n)$ are the unit-norm eigenvectors of the covariance matrix $\mathbf{C}_\mathbf{x} = E\{\mathbf{x}\mathbf{x}^T\}$, and $\mathbf{D} = \mathrm{diag}(d_1, \dots, d_n)$ is its diagonal matrix of the eigenvalues of $\mathbf{C}_x$.

The ICA can be computed in many ways, (see Hyvärinen et al., 2001, for further references). FastICA (Hyvärinen and Oja, 1997) is a fast fixed point algorithm, which is used in the publications of this thesis. The FastICA algorithm estimates the model in two stages. The first step reduces dimensionality and whitens the data. The second step finds a rotation that maximizes the statistical independence of the components. The dimensionality reduction and decorrelation step can be computed, for instance, with principal component analysis or singular value decomposition.

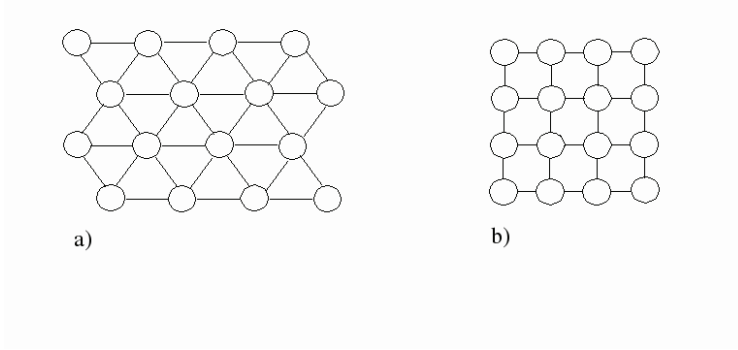


Figure 4.1. Neighborhood topologies of the Self-Organizing Map: a) hexagonal and b) rectangular.

Text data and ICA

It is assumed that the independent components are nongaussian. In earlier ICA research for text data, it is noted that while natural signals based on sensory data are typically nongaussian, text data is not natural in a similar sense, as it is based on an encoding process (Honkela et al., 2010). Nevertheless, word contexts seem to be nongaussian, because the data is sparse with a large probability mass for values close to zero, but with a heavy tail, in other words it is Zipfian (Section 2.1.1).

4.6.5 Self-Organizing Map

The Self-Organizing Map is an unsupervised artificial neural network model (Kohonen, 2001). It is useful for visualization of high-dimensional data, as it produces a topological ordering onto a low-dimensional grid of prototype vectors.

The map consists of a number of map units, or prototype vectors, that are of the same dimensionality as the input data. The map units are organized on a grid. The organization of the grid is either rectangular or hexagonal. In a rectangular organization, each map unit except for those at the edge of the map have four neighbors, and in hexagonal ordering, each non-edge map unit has six neighbors, see Fig. 4.1. The map organization can either be a sheet with edges, or a toroid with no edges.

The SOM is trained according to the competitive learning principle. For each input vector, a prototype vector that best matches the input is selected. The Best Matching Unit (BMU) is the prototype vector with the smallest distance to the input vector in some metric, usually Euclidean distance. In case of only partial information, BMU is searched by the ex-

isting part. The BMU and its neighboring prototype vectors in the topological ordering are adapted toward the input. The adaptation rule is typically expressed as:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{ci}(t)[\mathbf{x}(t) - \mathbf{m}_i(t)], \tag{4.29}$$

in which the c is the index of the BMU. $h_{ci}(t)$ is the neighborhood function defining how large the neighborhood is, $\mathbf{m}_i$ is the i-th prototype vector, $\mathbf{x}(t)$ the input vector, and t is a discrete time coordinate. Most commonly used is the Gaussian neighborhood function:

$$h_{ci}(t) = \alpha(t) \cdot \exp\left(\frac{||\mathbf{r}_c - \mathbf{r}_i||^2}{2\sigma^2(t)}\right), \tag{4.30}$$

where $0 < \alpha(t) < 1$ is the learning-rate factor monotonically decreasing in the course of the learning and $\sigma^2(t)$ corresponds to the neighborhood radius, also decreasing monotonically in the course of the learning. The $\mathbf{r}_c$ and $\mathbf{r}_i$ are vectorial locations prototypes m_c and m_i on the grid. Another commonly used neighborhood function is a 'bubble' which is constant in the whole neighborhood and zero elsewhere. In the beginning of the learning process, the neighborhood size and α are relatively large to obtain first a rough, global ordering. The value decreases during the process to achieve a more local ordering. In practice, typical value for α is close to (but smaller than) 1 and the neighborhood size may be half the size of the diameter of the map for the Gaussian function in the beginning of learning (Kohonen, 2001).

The initialization of the prototype vectors can be random, taken from the available input samples, or sampled from the two largest principal component eigenvectors of the data. The latter case is the recommended one. The SOM can be trained either in sequential or in batch mode. In sequential training the prototype vectors are adapted after each input to the SOM, as described earlier. Batch training is a faster alternative (Kohonen, 2001). In batch training, the whole data set is presented to the map before the map prototype vectors are adapted at all. In each training step, the data is then partitioned according to the Voronoi regions of the map weight vectors. Next, the new prototype vectors are calculated as

$$\mathbf{m}_i(t+1) = \frac{\sum_{j=1}^{n} h_{ci}(t)\mathbf{x}_j}{\sum_{j=1}^{n} h_{ci}(t)}, \tag{4.31}$$

where $c = \arg min_k\{\| \mathbf{x}_j - \mathbf{m}_k \|\}$ is the index of the BMU of the data sam-

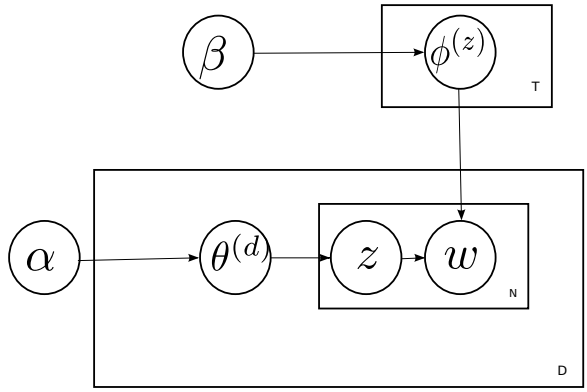


Figure 4.2. A plate diagram illustrating the LDA model

ple $\mathbf{x}_j$. The new prototype vector is a weighed average of the data samples, and the weight for each data sample is the neighborhood function value $h_{ci}(t)$ at its BMU c. In information visualization, the SOM is trustworthy, and it is comparable to the best methods for precision, but not in terms of recall (Nybo et al., 2007).

4.6.6 Topic models and Latent Dirichlet Allocation

Generative topic models are based on the idea that documents are a mixture of topics and that they are generated from latent random variables. Documents are then fitted to find the best set of latent variables that can explain the observed data (Steyvers and Griffiths, 2007). The probabilistic topic modeling was introduced as Probabilistic Latent Semantic Indexing (pLSI) (Hofmann, 1999), and further extended as Latent Dirichlet Allocation (Blei et al., 2003). The account describing the relationship of these models to PCA and ICA is given by Buntine and Jakulin (2006). These models have been explicitly developed to model count data, and they make assumptions about the distributions in different levels. See, for example, Newman et al. (2011) and Du et al. (2012) for recent advanced applications. Topic modeling has mostly concentrated on document representations, but some research has also been carried out with a short context around the target word (Brody and Lapata, 2009; Chrupała, 2011).

Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is a generative probabilistic model for discrete data, such as text corpora. It is based on an idea that documents are represented as random mixtures over latent (hidden) topics. Each topic then has a distribution over words. It is a three-layer

model, where the topics are sampled from a topic distribution, and then single words are sampled from the chosen topic. A plate diagram of the LDA model is shown in Figure 4.2. Only the words are observed, all the other parameters are hidden.

More specifically, a generative process for each word w in a corpus D of M documents is assumed. For each document d, that is, a sequence of words $d = (w_1, \ldots, w_N)$, the length of the document N is drawn from a Poisson distribution $N \sim \text{Poisson}(\xi)$. Let us denote the topics with a fixed dimensionality T of the Dirichlet distribution and the dimensionality of the topic variable z. The probability to sample the jth topic for the ith word w_i is $p(z_i = j)$, and $\phi^j = p(w|z_i = j)$ is the multinomial distribution over words for topic j (Griffiths et al., 2007). For each word w_i in the document, the topic z_i is drawn from $p(z_i|\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is in turn drawn from a Dirichlet distribution with a parameter $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_t)$, $\boldsymbol{\theta} \sim \text{Dir}(\boldsymbol{\alpha})$. α_j can be interpreted as the prior observation count for the number of times topic j is sampled in each document, before having observed any actual words from that document. In addition, it is convenient to use a symmetric Dirichlet distribution with a single parameter $\alpha = \alpha_1 = \ldots = \alpha_T$ (Griffiths et al., 2007). Finally, a word w_i is chosen, on the condition of the chosen topic z_i, from $p(w_i|z_i, \beta)$. It is also possible to use a symmetric β prior on ϕ, which is used, for example, in Griffiths and Steyvers (2004). Griffiths et al. (2007) suggest values $\alpha = 50/T$ and $\beta = 0.01$, which are reported to work reasonably well.

The model makes a number of simplifying assumptions, especially, the dimensionality T of the Dirichlet distribution is assumed to be known and fixed, and N is independent of the other data generating variables z and θ and the Poisson assumption of the document length is not critical.

The central problem is to obtain the posterior distribution $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, where $\mathbf{w}$ is the set of N words $(w_1, \ldots, w_N)$, $\mathbf{z}$ is the set of topics or hidden variables, and $\boldsymbol{\theta}$ gives their probabilities, but the problem is not computable in general (Blei et al., 2003).

Gibbs sampling

The distributions cannot be calculated as such but they need to be estimated with, for example, Variational EM (Blei et al., 2003) or Gibbs Sampling (Griffiths and Steyvers, 2004). The Gibbs sampling algorithm is an algorithm in the form of Markov chain Monte Carlo, which is a family of methods designed for sampling values from complex distributions

(Griffiths et al., 2007). The Gibbs sampling applied to topic modeling carries out the actual topic extraction by directly estimating the posterior distribution over z given the observed words d, marginalizing out ϕ and θ (Griffiths and Steyvers, 2004).

$$P(z_i = j|\mathbf{z}_{-i}, w_i, d_i, \cdot) \propto \frac{C^{WT}_{w_ij} + \beta}{\sum_{w=1}^{W} C^{WT}_{wj} + W\beta} \frac{C^{DT}_{d_ij} + \alpha}{\sum_{t=1}^{T} C^{DT}_{d_it} + T\alpha} \tag{4.32}$$

C^{WT}_{wj} contains the number of times word w is assigned to topic j, and C^{DT}_{dj} contains the number of times topic j is assigned to some word token in document d excluding the current instance i in both cases. In reality, the equation is further divided for topic k by the sum over all topics.

The sampling algorithm starts by assigning each word token to a random topic z_j with $j = 1 \dots T$. For each word token, an entry that corresponds to the current topic assignment is taken out of the count matrices. Then a new topic is sampled from the distribution in Eq. (4.32) and count matrices are incremented with the new topic assignment. Each Gibbs sample thus consists of the set of topic assignments to all N word tokens in the corpus. In the beginning, the Gibbs samples are a poor estimate of the posterior, but after many iterations, the approximation improves.

LDA for short context

LDA-related approach can be also used in a short context around the target word. Brody and Lapata (2009) use a method related to LDA in a sense induction task. The model operates on what they call a local context, a small context around the target word, instead of a global topic around a document. $P(s)$ is used as a distribution over senses of an ambiguous target word in a context window, and $P(w|s)$ for the probability distribution over context words w given a sense s. The model generates each word w_i in the context window by first sampling a sense from the sense distribution, and then choosing a word from the sense-context distribution. All the other variables except for the word itself are hidden. The model specifies a distribution over words within a context window:

$$P(w_i) = \sum_{j=1}^{S} P(w_i|s_i = j)P(s_i = j), \tag{4.33}$$

where S is the number of senses. It is assumed that each target word has C contexts and each context c consists of N_c word tokens.

Table 4.5. The correspondence of the LDA model and the Chrupała model of the word classes

LDA	Chrupała model
Topics	Word classes
Documents	Word types
Words	Context features

In another related work, Chrupała (2011) uses a similar model with a small context of a single word to the left and a single word to the right around the target word. The model is tested in several statistical NLP tasks: named entity recognition, morphological analysis and classification of semantic relations. In this model, a word in the vocabulary corresponds to a document in the LDA model, a word corresponds to a context feature, and a topic corresponds to a word class, see Table 4.5. In the generative model, the number of topics T from the LDA model corresponds to the number of latent classes, D is the vocabulary size, N_d the number of left and right contexts in which word type d appears, z_{n_d} is the class of the word type d in the n_dth context and f_{n_d} is the n_dth context feature of word type d. The model provides two types of word representations once trained: Each θ_d gives the latent class probability distribution given a word type and each ϕ_k gives a featured distribution given a latent class (Chrupała, 2011).

4.6.7 Neighbor Retrieval Visualizer

The Neighbor Retrieval Visualizer (NeRV) used in Publication III is an unsupervised dimensionality reduction and visualization method (Venna and Kaski, 2007; Venna et al., 2010). The method approaches the visualization problem as an information retrieval problem. The input space P is the high-dimensional original space, where neighbors for a sample x_i can be defined in P_i. The neighbors can be, for example, defined through a fixed neighborhood radius or be a fixed number of nearest neighbors. Then output space Q is examined, and the number of original neighbors of x_i that are retained in the neighborhood Q_i are checked. In addition, the amount of error in the form of false positives and false negatives is checked using information retrieval concepts of Precision and Recall (Eqs. 4.16 and 4.17). The method defines a cost function to the classification errors introduced in the dimension reduction: $E = N_{fp}C_{fp} + N_{fn}C_{fn}$, with an associated cost for both types of errors.

The model is further extended from binary neighborhood to probabilistic

neighborhood, where both the input and output neighborhoods are probabilistic. In addition, the binary Precision and Recall are replaced by probabilistic measures.

In the output space, the probabilistic model of retrieval is defined as

$$q_{j|i} = \frac{\exp\left(-\frac{\|\mathbf{y_i}-\mathbf{y_j}\|^2}{\sigma_i^2}\right)}{\sum_{k\neq i}\exp\left(-\frac{\|\mathbf{y_i}-\mathbf{y_k}\|^2}{\sigma_i^2}\right)}, \tag{4.34}$$

where $\|\mathbf{y}_i-\mathbf{y}_j\|$ is the Euclidean distance and $1/\sigma_i$ is a constant that allows the function to grow at an individual rate for each point i studied. The Euclidean distance $\|\mathbf{y_i}-\mathbf{y_j}\|^2$ can be replaced with any suitable difference measure in the original data. The probabilistic model of relevance, $p_{j|i}$ is defined analogously.

The Kullback-Leibler divergence $D(p_i, q_i)$ is the generalization of Recall, and $D(q_i, p_i)$ is the generalization of the Precision and

$$D(p_i, q_i) = \sum_{j\neq i} p_{j|i}\log\frac{p_{j|i}}{q_{j|i}}. \tag{4.35}$$

The NeRV optimizes directly a cost function which incorporates Precision and Recall:

$$\begin{aligned} E_{NeRV} &= \lambda E\{D(p_i, q_i)\} + (1-\lambda)E\{D(q_i, p_i)\} \\ &\propto \lambda\sum_i\sum_{j\neq i} p_{j|i}\log\frac{p_{j|i}}{q_{j|i}} + (1-\lambda)\sum_i\sum_{j\neq i} q_{j|i}\log\frac{q_{j|i}}{p_{j|i}}. \end{aligned} \tag{4.36}$$

The choice of λ defines whether minimizing false positives or false negatives is more important in a given case and it is left to the user. Then the cost function is optimized to minimize the total cost. Taking the example from (Venna et al., 2010), if $\lambda = 0.1$, the error in Precision is nine times as expensive as an error in Recall.

5. Analysis of the similarities of languages using unsupervised methods

This chapter starts the discussion of similarity at the highest level of analysis: the complexity of natural languages, which is the theme of Publication I. It is generally assumed that languages are equally expressive (Sapir, 1924), that is, they can convey the same meaning. Another standard assumption related to the previous one is that the human languages are equally complex, even though the languages can differ in the distribution of their complexity: A simple morphology could be evened out by more complex syntactic distinctions or lexical differences (Bane, 2008). This assumption is without definite proof, though. See, for example, Shosted (2006) for discussion. Whether the higher complexity on one level is compensated by lower complexity on another level or not, different languages use different structures to convey meaning. For example, Miestamo et al. (2008) consider various linguistic points of view related to complexity. This complexity of different levels of representation represents challenges, for example, in machine translation.

In this chapter, the focus is on analyzing the complexity of 21 European languages using data-driven methodologies: conducting separate analyzes on morphological, and syntactic level, which are further compared to linguistic analyzes of the syntactic complexity. Kolmogorov complexity, and its approximation using compression were introduced in Section 4.2.1, and the compression-based methodologies are the basis of the experiments of this chapter. In addition to the compression experiments, morphological complexity was also analyzed using an unsupervised morphology analysis method, Morfessor (Creutz and Lagus, 2002), introduced in Chapter 4.

Publication I uses three different methods to analyze the similarities and differences of languages. The first method (Juola, 1998, 2008) utilizes compression in a *single language* at a time, comparing the difference in

the size of the original compressed document and the size of a document after a *transformation* is applied to the document. Each transformation destroys the information at a certain linguistic level, and the effect of the transformation can be analyzed.

The second approach also uses compression in analysis, now measuring the *pairwise* similarities of languages. The methodology compares the sum of the lengths of two separately compressed documents to the length of a compressed document which is a concatenation of the two documents. This approach does not concentrate on any single linguistic level of language, but rather attempts to measure the overall similarity of the languages.

The third approach concentrates on the morphological level of language. It differs from the two previous approaches by the choice of methodology: an unsupervised morphology analysis method Morfessor is used to obtain features used in the similarity comparison.

5.1 Text as data

To analyze the similarities of different languages, the text data used in the analysis should be aligned or at least a translation of the same text. For this purpose, Juola (1998) uses the Bible which exists in a large number of languages in electronic form, whereas Cilibrasi and Vitányi (2005) use the texts of the Universal Declaration of Human Rights in 35 languages and Benedetto et al. (2002) texts retrieved from the European Union archives. In Publication I, we have used the texts of the European constitution, available in the 21 official languages of the European Union at the time of writing (2005), see Table 2.2 for the list of languages and their families. All of the texts except Greek are written with the Latin alphabet or its local extensions (i.e. including letters such as ä or ø), and the files were encoded with the UTF-8 encoding.

Another possibility could have been the Europarl corpus (Koehn, 2005), which currently contains texts in 21 languages (excluding Irish and Maltese but including Bulgarian and Romanian). None of these resources represent common everyday language. The texts in the Bible are somewhat archaic, and the rest of the resources are of legal genre, which may not be considered the most characteristic of a given language. However, these are examples of these natural languages and retain syntactic and morphological characteristics of these languages, despite their limitations.

5.2 Analysis of morphological and syntactic level through compression

Juola (1998) proposed using Kolmogorov complexity and its approximation with compression to study the complexity of language. He showed that the variation of length in uncompressed text was longer than in the compressed text. He also suggested distortion as a measure of the effect of the morphological level (Juola, 2005, 2008). The transformation method will be discussed in more detail in the following.

5.2.1 Transformation method

The basic idea is that transformation at a certain linguistic level distorts the information at that level, and the effect of the transformation in compression can be used as an indicator of the complexity of the linguistic level the distortion was applied to (Juola, 2005). For example, the morphological information can be used in compression: The program tries to find units that can be reused, for example, parts of words. This means that words *surf*, *surfs*, *surfer*, and *surfing* can be encoded with the root *surf*, and a variable ending, which reduces the coding length. We expect that permutation of linguistic units at the morphological level should change the coding length and the amount of change can be then measured when compared to the original length. In Publication I, Kolmogorov complexity is approximated by compression with a block compression method *bzip2* based on the Burrows-Wheeler algorithm (Burrows and Wheeler, 1994).

To analyze the complexity at the morphological level, the morphological information was hidden by replacing each word type in each document by a random number in the range $[10\,000 \ldots 30\,000]$. This method was proposed by Juola (1998). In that article, the transformation was carried out by using numbers within the range of $[1 \ldots 31\,000]$. In Publication I, we opted at using the numbers of same length, instead of varying length at the transformation. This transformation hid any information between related word types.

The value for each language was obtained by

$$\mathrm{C}_{morph}(l_i) = \frac{V_{orig}(l_i)}{V_{morph}(l_i)}, \tag{5.1}$$

where $V_{orig}(l_i)$ is the size of the compressed original document in bytes and $V_{morph}(l_i)$ the size of the altered document in bytes for each language

l_i.

A compression program can also use word order information, i.e. finding words that occur in the similar order multiple times. Randomizing the word order in a sentence breaks this information, and allows an analysis of the complexity of the syntactic level. Again, the complexity values C_{synt} were obtained by dividing the size of the original compressed V_{orig} by the size of the altered compressed document V_{synt}.

$$\mathrm{C}_{synt}(l_i) = \frac{V_{orig}(l_i)}{V_{synt}(l_i)}, \tag{5.2}$$

Finally, in Publication I the morpho-syntactic complexity measure is calculated by summing the file sizes for the morphologically and syntactically altered files and using the sum to divide the size of the original file.

$$\mathrm{C}_{ms}(l_i) = \frac{V_{orig}(l_i)}{V_{morph}(l_i) + V_{synt}(l_i)}. \tag{5.3}$$

Juola (2005) suggested the randomization of the word order to measure syntactic complexity. In a follow up (Juola, 2008), published around the same time as Publication I, another method is used instead with the Bible verses as data: random deletion of ten percent of letters for morphological complexity analysis; random deletion of ten percent of words for syntactic level analysis; and random deletion of ten percent of verses for pragmatic analysis.

5.2.2 Results

The results for the morphological, syntactic and morpho-syntactic analysis are shown in Fig. 5.1. On x-axis, the languages are ordered by their complexity score, with the values of $C(l_i)$ on y-axis. The languages with the lowest complexity score were the same in each case, even though the order differed slightly. These were the Romance languages Italian (it), Spanish (es), French (fr), and Portuguese (pt); Celtic language Irish (ga); and the Slavic language Slovene (sl).

In the high complexity end of this measure, there was more variation. Finnish (fi) was among the top three in each case, German (de) had top-two scores in morphological and morpho-syntactic scores, and had a fairly high score in syntactic complexity. The authors suggested in Publication I that the legalese type of language augments the score in German due to compounding. Polish (pl) also gets fairly high complexity scores. The

scores for Estonian (et) are also high for morpho-syntactic and syntactic complexity, but lower for morphological complexity.

Figure 5.2 visualizes the languages using word-order information V_{synt} on x-axis and morphological complexity V_{morph} on y-axis, aggregating the results of of Figs. 5.1a and 5.1b. It shows that based on this measure, the Romance languages have both a low syntactic and morphological complexity score, whereas Finnish is complex at both levels.

5.2.3 Comparison to linguistic classification

As an evaluation, the results obtained using the compression method were compared to a linguistic analysis. One natural approach is to compare the results to the language families, shown in Table 2.2. Another resource is the collection of syntactic flexibility values defined by Bakker (1998). The values describe the flexibility of the word order of a language based on ten factors, such as the order of the verb and objective, order of the adjective and its head noun or order of the genitive and its head noun. Based on these factors, a single value in the range of $C_{Bakker} = [0, 0.1, 0.2, \ldots, 1]$ has been defined for each language. Small values indicate an inflexible word order, and large values a more flexible word order.

The results obtained with the compression method were compared to Bakker's values, shown in Table 5.1. For this dissertation, the Spearman rank correlation (Spearman, 1904) between them was calculated. It was found that there is a significant dependence between these values. The Spearman's rank correlation coefficient is $\rho = 0.65$, with $p = 0.0026$, which is significant as it is over the critical value of $\rho > 0.584$ for 19 samples at the two-tailed p-level of $p < 0.01$.

5.3 Pairwise similarity of languages by compression

Cilibrasi and Vitányi (2005) propose another compression-based approach where the the similarity of languages is calculated using pairwise comparisons. They define a distance metric, Normalised Compression Distance (NCD), which is used to obtain a pairwise distance matrix. This distance matrix can be then clustered into a language tree. They obtain results which are in line with the current linguistic knowledge about language families. This approach measures overall similarity, and does not consider any differences in different levels of analysis. Benedetto et al. (2002) also

Table 5.1. Bakker's flexibility values (Bakker, 1998) vs. compression results. Czech and Hungarian were not included in Bakker's study, and thus left out of this table. Adapted from Publication I.

Bakker			Compression		
rank	language	flexibility	rank	language	syntactic complexity
1	fr	0.10	1	fr	0.66
2	ga	0.20	2	es	0.68
3	es	0.30	3	pt	0.68
4	pt	0.30	4	ga	0.69
5	it	0.30	5	it	0.69
6	da	0.30	6	en	0.69
7	mt	0.30	7	sl	0.71
8	lt	0.30	8	nl	0.71
9	en	0.40	9	mt	0.72
10	nl	0.40	10	da	0.72
11	de	0.40	11	el	0.73
12	sv	0.40	12	sv	0.75
13	et	0.40	13	lv	0.75
14	sl	0.50	14	de	0.75
15	lv	0.50	15	pl	0.76
16	sk	0.50	16	lt	0.76
17	el	0.60	17	sk	0.77
18	pl	0.60	18	et	0.78
19	fi	0.60	19	fi	0.79

suggest the use of data compression in a similar fashion to build language trees.

In Publication I, an experiment was carried out following Cilibrasi and Vitányi (2005). A size of the compressed text file was again used as a measure of the Kolmogorov complexity as earlier. The basic principle behind the pairwise analysis is this: The compression program learns the characteristics of a language while processing it. If the language changes, the program needs to unlearn the old characteristics and learn the new language characteristics. If languages are similar, perhaps only small modifications to the old coding are needed.

Li et al. (2004) propose the use of such a compression distance as a normalized similarity metric applicable in different domains. In Publication I, the similarity score used in Publication I was

$$C(l_i, l_j) = \frac{V(l_i, l_j) - V(l_i)}{V(l_j)}, \tag{5.4}$$

where $V(l_i)$ is the size of the compressed file in language l_i, $V(l_j)$ the size of the compressed file in language l_j, and $V(l_i, l_j)$ is the length of the compressed file of the concatenated texts in l_i and l_j. Li et al. (2004) recommend a symmetric similarity score instead, replacing $V(l_i)$ in the numer-

ator by $\min(V(l_i), V(l_j))$ and $V(l_j)$ in the denominator by $\max(V(l_i), V(l_j))$. We hypothesized that in a pairwise comparison of languages, the relation might be asymmetric and language l_i might be better explained by language l_j, while this relation might not hold in the opposite direction. Cilibrasi and Vitányi (2005) produce a clustering from the matrix of pairwise similarities and build a dendrogram.

In Publication I, a Self-Organizing Map based visualization of the distance matrix was created (Fig. 6 in Publication I). The overall results were poor, even though there were some language pairs close to each other that also belong to same language family. Using a symmetric distance measure did not change this result, either. A recent review revealed that an error was made during the experiment, not known at the time the original experiments were run. Cebrián et al. (2005) point out that when using block or window based compressors such as bzip2, the concatenated file should fit into a single block, as the model begins anew at the beginning of each new block. When using the bzip2 compressor, the maximum block size is thus 900 kilobytes. In the experiment of Publication I, most of the text files used in the experiments exceeded the block size slightly, making the length of the concatenation over double the block size, thus rendering the results worthless.

In this dissertation, the experiment was repeated with smaller text fragments that fit into this constraint introduced by bzip2. The analysis was thus limited to Parts I and II of the constitution text in each of the languages. The pre-processing steps were otherwise the same as in Publication I. The resulting pairwise similarity matrix is visualized using the two first components of the SVD, shown in Figure 5.3. Now the results look more convincing. The Romance (ro) languages form a group in the lower right, and there are some pairwise similarities, such as the Slavic (sl) languages Czech and Slovak and the Germanic (ge) languages Danish and Swedish. In the middle left in the figure there is a cluster of languages which seem fairly similar with this measure. The cluster contains most Germanic languages (except English), the Baltic languages Latvian and Lithuanian, and the Fennic languages Estonian and Finnish. This visualization shows Greek as an outlier in the top right corner. This result is also natural, as Greek has its own alphabet, which makes it most dissimilar from the rest of the languages at the surface level. This feature presents a further complication in analyzing the pairwise similarity of languages written with a different alphabet. There are several words

that have been borrowed from Greek to other languages, but the use of the alphabet (and subsequent encoding) hides efficiently this relation between languages. It could only be taken into account, if a translation to a shared alphabet (phonetic, for example) was used.

5.4 Analysis based on unsupervised morphological analysis

In Publication I, Morfessor (see Chapter 4.6.1) analysis was also carried out for the 21 European languages.The analysis was carried out for each of the 21 languages with a hypothesis that a high(er) number of retrieved morphs indicates a rich morphology. Many of the languages had not been processed with Morfessor before. The fairly short length of the documents limits the quality of the morph segmentation.

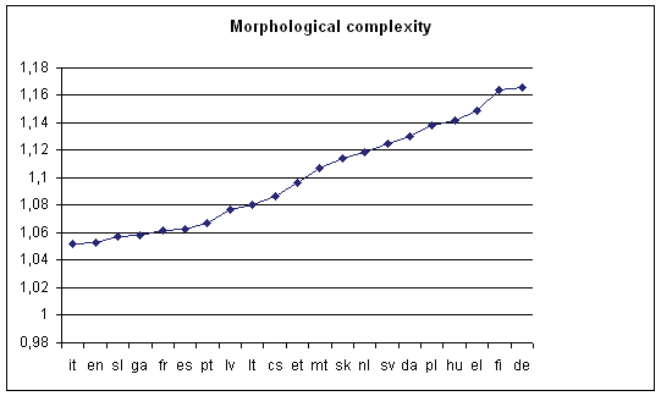
The work included calculating seven different variables from each language sample: i) number of types in the sample, ii) number of tokens in the sample, iii) average number of morphs per word, iv) the mean, v) variance, vi) skewness and vii) kurtosis of the morph length distribution. Vectors containing these variables were normalized based on the variance, and the results were visualized with the Self-Organizing Map, shown in Figure 5.4. The Romance languages and English at the top of the map separate again fairly well from the rest. At the bottom, Hungarian and Estonian are close together, with Finnish further down. In addition, for example, Czech and Slovak are very close to each other, as are German, Danish and Swedish. In addition to the general u-matrix visualization, the effect of each variable to the organization of the map can be evaluated, see Fig. 5.5. The values of average skewness and kurtosis of the morph length distribution separate Dutch from other languages. Analysis of the morph length distribution does not show large differences between related languages such as German and English (Figure 5.6) besides the fact that in Dutch, there are a larger number of morphs of lengths $|m| = 2$ and $|m| = 3$ in comparison to most languages except English.

In Publication I, it was remarked that the ranking of the languages based on the average number of morphs per language resembles the ranking obtained by the compression of the morphological level, (see Section 5.2). In this dissertation, the association between the two is further confirmed using the Spearman's rank correlation coefficient. For these rankings, $\rho = 0.61$ with a $p = 0.0033$ is obtained, which is significant and over the critical value of $\rho = 0.556$ for 21 samples at the two-tailed analysis

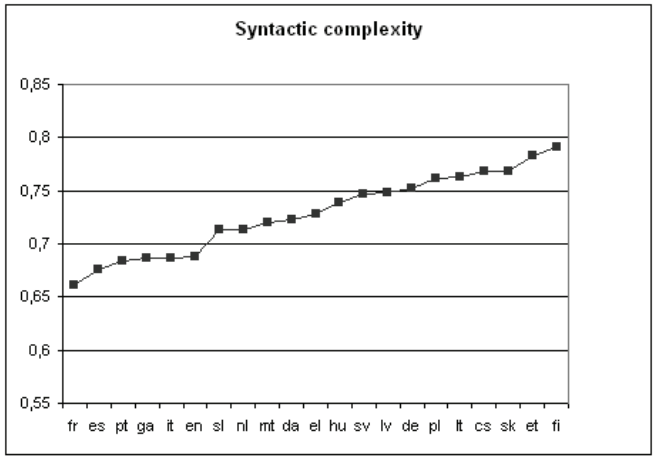
with a p-level of $p < 0.01$. The results are not surprising, as rich morphology in a language also manifests itself as a large number of morphs in a language.

Table 5.2. Compression of morphological complexity vs. average number of morphs per word

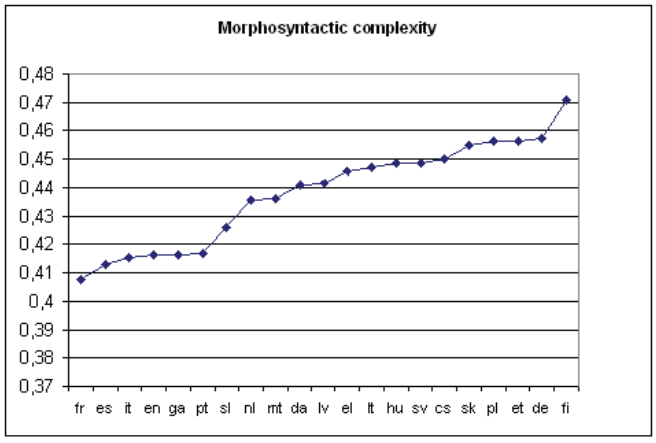
Compression			Morfessor		
rank	language	morph complexity	rank	language	avg. morph length
1	de	1.17	1	et	1.16
2	fi	1.16	2	fi	1.15
3	el	1.15	3	hu	1.15
4	hu	1.14	4	sk	1.15
5	pl	1.14	5	cs	1.14
6	da	1.13	6	el	1.12
7	sv	1.13	7	pl	1.11
8	nl	1.12	8	lt	1.10
9	sk	1.11	9	sl	1.10
10	mt	1.11	10	de	1.09
11	et	1.10	11	mt	1.09
12	cs	1.09	12	lv	1.09
13	lt	1.08	13	sv	1.09
14	lv	1.08	14	da	1.08
15	pt	1.07	15	ga	1.06
16	es	1.06	16	nl	1.05
17	fr	1.06	17	pt	1.05
18	ga	1.06	18	es	1.04
19	sl	1.06	19	it	1.04
20	en	1.05	20	fr	1.04
21	it	1.05	21	en	1.03



(a) Morphological complexity



(b) Syntactic complexity



(c) Morphosyntactic complexity

Figure 5.1. The European languages ordered based on the compression results at different levels of analysis. a) morphological complexity b) morphosyntactic complexity and c) syntactic complexity, from Publication I

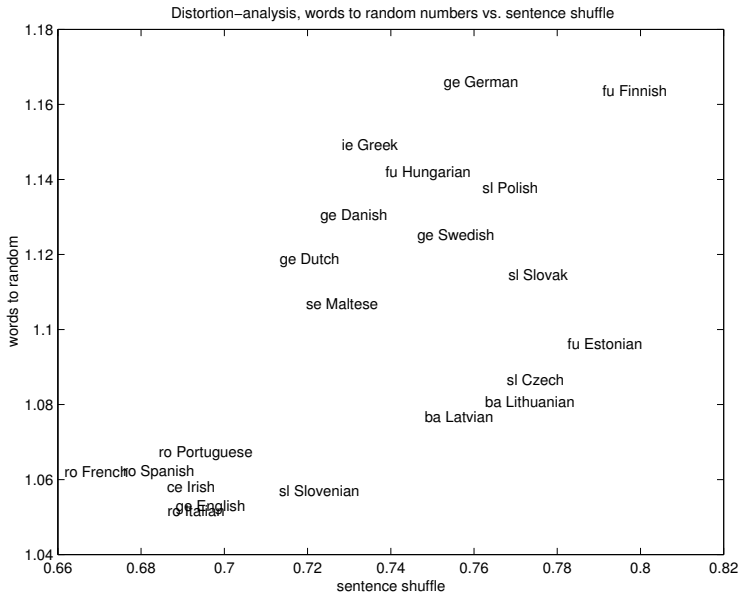


Figure 5.2. Map of the languages: morphology vs. word order information. From Publication I.

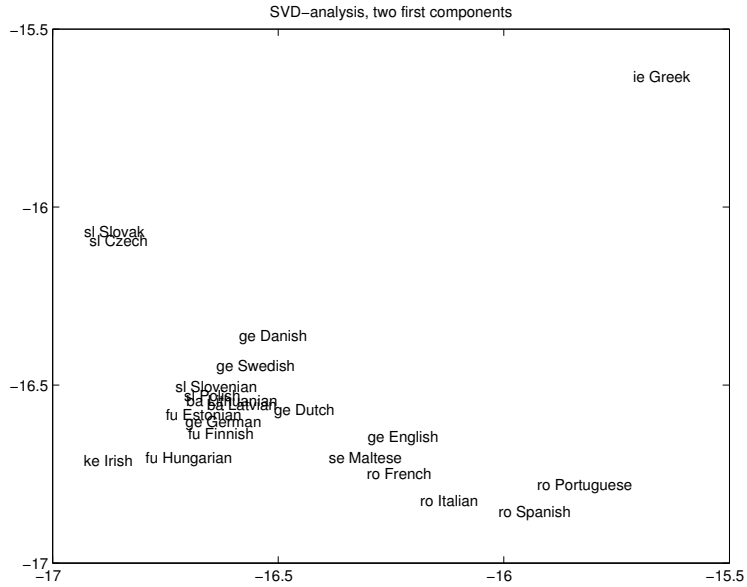


Figure 5.3. Visualization of the pairwise similarity comparisons of the European languages using the two first components of the SVD

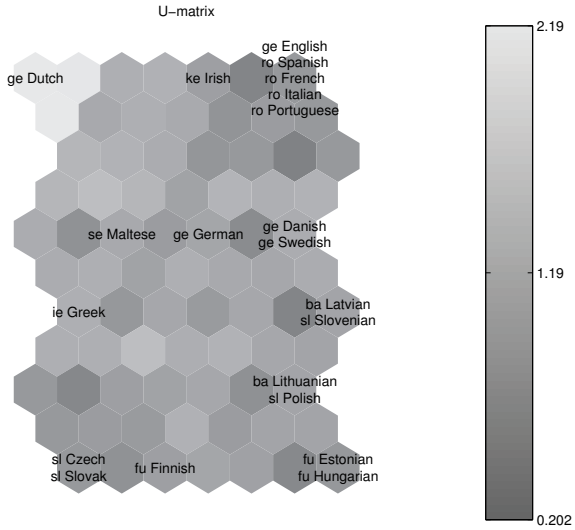


Figure 5.4. Visualization of the languages with the Self-Organizing Map using the Morfessor-induced features. From Publication I

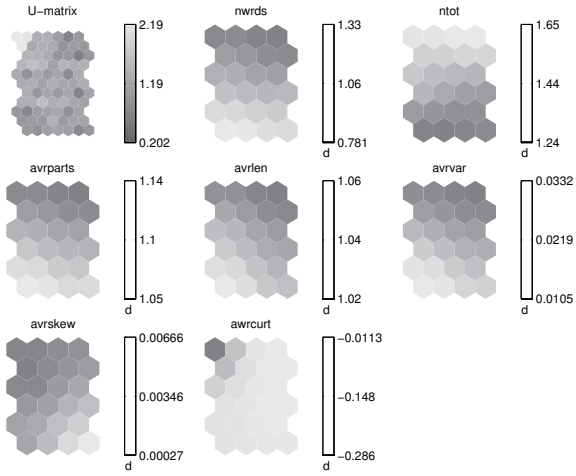
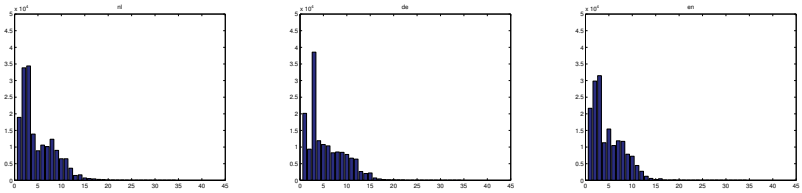


Figure 5.5. Distribution of the variables of the Morfessor experiment on the map. From Publication I



(a) Dutch **(b)** German **(c)** English

Figure 5.6. Comparison of the morph length distributions for Dutch, German and English

6. Sense in vector space models: similarity, interpretable components and exploration

In this chapter, the focus changes from the analysis of the similarity of different natural languages into representing similarity of different words in one language. The method choice is the word vector space model (VSM), for which the basics were introduced in Chapter 4. The choice of methodology is focused on unsupervised methods without strong linguistic assumptions instead of supervised learning (classification).

This chapter first introduces semantic evaluation data sets and methods based on the work in Publication III and Publication IV. A bilingual VSM representation using Independent Component Analysis (ICA) is also introduced based on Publication II. Then the focus moves to the use of the ICA for finding semantic information from word space models.

The first contribution of this chapter is the use of unsupervised methods to find sets of semantically related words. The word sets found with two methods, ICA and Latent Dirichlet Allocation (LDA) are compared to category labels of two fairly large semantic dictionaries. The second contribution is a further analysis of the categories: it was shown that the categories of the evaluation sets have differences and some categories are more difficult to represent with the word vector space models than others. The third contribution in this part is the demonstration on how to use the visualization methods as a tool for exploration in analysis of the categories and the relations between categories. In addition, a methodology was presented to find frequent sets of related words using ICA or LDA including a qualitative analysis of the type of the word sets found.

6.1 Evaluation of word vector space models

In order to compare the different methodological alternatives, the quality of the vector space models must be evaluated. The evaluation meth-

ods can be divided into direct (intrinsic) and indirect (extrinsic) methods (Sahlgren, 2006; Suominen et al., 2008). In indirect evaluation, the models are evaluated through their performance in some NLP or text mining task. Such a task can be, for example, information retrieval for document models or word sense disambiguation for word space models. For a summary of evaluation measures in text mining tasks, see for example Suominen et al. (2008). An observation of the fitness of a model is made indirectly. The problem of such an approach is that the fitness might not generalize for other tasks (Virpioja et al., 2012). In direct evaluation, the fitness of the vector space model is evaluated through separate test sets directly. Based on research in psychology and cognitive science, data sets that list words that are judged similar have been created for direct semantic evaluation purposes.

6.1.1 Semantic relatedness

Similarity judgment is considered to be one of the most central functions in human cognition (Goldstone, 1994). It is used to store and retrieve information, and to compare new situations to similar experiences in the past. Category learning and concept formation also rely on similarity judgments (Schwering, 2008).

Semantic relatedness is a wider term encompassing different relation types between words (Budanitsky and Hirst, 2006). The concept of semantic similarity is often used to mean semantic relatedness. Different types of semantic relatedness can be defined, such as synonymy (*automobile–car*), antonymy (*good–bad*), hypernymy (*vehicle–car*) and meronymy (*car–wheel*). A special case is a category (VEHICLE–*car, bicycle*,...), in which the members of the category are perceived to share similar characteristics.

6.1.2 Measuring similarity

According to Schütze (1993), "in a vector space, related words are close to each other, unrelated are distant". Different evaluation schemes have also been devised to measure the relatedness. Often, Precision (Pr) (Eq. 4.19) is used as the measure of the quality of the vector space model. We can also define Error (Err) as

$$Err = 1 - Pr. \tag{6.1}$$

The similarity of the vector representations is carried out by calculating their difference based on some distance or similarity metric. The most commonly used distance and similarity metrics have been listed in Table 4.3. Using cosine distance in the similarity calculation captures the idea that the angle between the vectors is more important than the length of the vectors and it is the most commonly used in VSM research (Turney and Pantel, 2010). It is also used in most of the Publications of this dissertation. The only exception is Publication III where Euclidean distance is used in the calculation when the SOM and NeRV are used.

Different methodologies for conducting semantic similarity evaluation have been proposed. In this section, the different methodologies are summarized, and in the following section, different evaluation test sets are introduced. First, the neighborhood can be analyzed directly by checking whether the local neighbors of a given target word are similar, that is belonging to a same class or category (Sahlgren, 2006). Calculating pairwise similarities for a large number of word vectors in a vector space model is a computationally demanding task, which is why the evaluation is usually carried out for the specific evaluation sets. In this case, Precision is the fraction of words out of N closest neighbors of cue word w_i that belong to the same class with the cue word, averaged over all cue words. If only the closest neighbor is examined for each cue word, the Precision is defined as the fraction of cases where the closest word has the same class label.

Second, comparisons between a semantically similar word pair and unrelated distractor words can be made. The distance between the cue word and its pair should be smaller than between the cue word and the distractor words: i.e. the system should be able to find the 'correct' answer among multiple 'choices'. In this case, Precision is defined as the fraction of times the system picks the correct answer.

Third, a category-based method of word clusters can be used (Bullinaria and Levy, 2007). In this method, the words in the vocabulary are grouped into different categories. For each category, a category centroid is calculated as a mean of the word vectors that belong to the category. For each word in the vocabulary, the distance to all the category centroids[1] is calculated, and the task is to check whether each word is closest to the category centroid it is labeled with. In this case, Precision is defined as the fraction of times the cue word is closest to the category centroid the word is

[1]The category centroid for the category the current target word belongs to is recalculated to exclude the target word.

classified to.

In each case, Precision is used as the measure of the performance: In how many cases did the system find the correct answer, or how many of the neighbors belong to the same class, or in how many cases was the cue word closest to the category centroid. In the following sections, different types of semantic evaluation sets used in VSM research are introduced in more detail.

6.1.3 Syntactic and semantic categories

The discussion of the evaluation sets will begin from sets that form categories, i.e. words that are judged similar in some sense. Syntactic categories are covered first, followed by semantic categories.

Syntactic categories

Syntactic category information can be used if the syntactic categories of the words are known. In Publication IV, two different syntactic category test alternatives are used. Patel et al. (1997) and Bullinaria and Levy (2007) use ten narrow syntactic categories using a category cluster based evaluation, whereas Sahlgren (2006) uses eight broad categories and analyzes the nearest neighbors for each word, checking whether they belong to the same syntactic category as the word. Honkela et al. (2010) do an analysis of the syntactic categories with the ICA components. They collected the word tags from the Brown corpus (Francis and Kucera, 1964), assigning a word category vector for each word with all the tags the word was labeled with.

In Publication IV, two similar sets to those used by Patel et al. (1997) and Sahlgren (2006) were created by tagging the 3 000 most frequent words with the part of speech tags of the Penn tree bank tag set (Marcus et al., 1993). Two different syntactic category test sets were used: Syncat 1 contained 10 narrow POS categories: Singular or mass noun (NN), Plural noun (NNS), Singular proper noun (NNP), Adjective in base form (JJ), Adverbial in base form (RB), Verb in base form (VB), Verb in past participle (VBN), Verb in -ing form (VBG) and Cardinal number (CD), with 50 words in each category. Syncat 2 contains seven broad POS categories: NOUN, VERB, ADJECTIVE, ADVERB, PREPOSITION, DETERMINER and CONJUNCTION with 20 words in each category. In both cases, the category cluster distance evaluation (Bullinaria and Levy, 2007) was used.

Table 6.1. The categories of the Battig set

Precious stone	Furniture	Sport	Vegetable
Unit of time	Fruit	Dance	Type of footgear
Relative	Weapon	Article of clothing	Insect
Unit of distance	Elective office	Part of a building	Girl's first name
Metal	Human dwelling	Chemical element	Male's first name
Reading material	Toy	Science	Flower
Military title	Country	Kind of money	Disease
City	Crime	Type of music	Tree
Kind of cloth	Carpenter's tool	Bird	Ship
Color	Type of fuel	Kitchen utensil	Fish
Nonalcoholic beverage	Substance for flavoring food	Weather phenomenon	Part of speech
Building for religious services	Member of the clergy	Musical instrument	
Four-footed animal	Occupation or profession	Alcoholic beverage	
Part of human body	Natural earth formation	Vehicle	

Battig

The Battig set[2] (Bullinaria and Levy, 2007) contains 53 categories with 10 words in each category, based on the 56 categories collected by Battig and Montague (1969). Two categories STATE and UNIVERSITY were left out as they contain too many compound words, and SNAKE category seems to contain many too polysemous words, such as *garden*, *black*, *garter*. The categories are listed in Table 6.1. The test set contains the words in each category in the order they are listed in Battig and Montague (1969). Only two words appear in two categories: *orange* in FRUIT and in COLOR, and *bicycle* in TOY, and in VEHICLE. The Battig set is used in Publications IV, and VI. In Publication IV, this set was called the Semcat set.

Other category-based evaluation sets not used in this dissertation include the ESSLLI 2008 set (Baroni et al., 2008), which contains 44 concrete nouns that belong to six classes, and 45 verbs that belong to nine semantic classes, Baroni's category list of 83 concepts in 10 categories (Baroni et al., 2010) based on an updated version of the Battig-Montague

[2]The word list is available on line from `http://www.cs.bham.ac.uk/~jxb/corpus.html`, accessed March 8, 2012

Table 6.2. The broad categories in the BLESS set with sample word from each set

Category	sample word
AMPHIBIAN_REPTILE	*alligator*
APPLIANCE	*dishwasher*
BIRD	*crow*
BUILDING	*castle*
CLOTHING	*blouse*
CONTAINER	*bag*
FRUIT	*apricot*
FURNITURE	*bed*
GROUND_MAMMAL	*bear*
INSECT	*ant*
MUSICAL_INSTRUMENT	*cello*
TOOL	*axe*
TREE	*acacia*
VEGETABLE	*beet*
VEHICLE	*ambulance*
WATER_ANIMAL	*carp*
WEAPON	*sword*

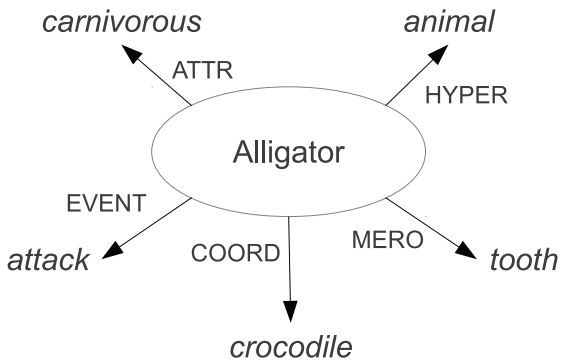


Figure 6.1. An example of the BLESS relations for a sample word belonging to the category AMPHIBIAN_REPTILE. From Publication VI.

list (Van Overschelde et al., 2004) and the Almuhareb list (Almuhareb, 2006), which contains 402 concepts.

BLESS

The Baroni-Lenci Evaluation of Semantic Spaces (BLESS) test set (Baroni and Lenci, 2011) is based on a body of earlier work on human similarity judgments. The data set contains 200 concepts in 17 broader classes or categories with 5–21 words per class. The classes are shown in Table 6.2 with an example word from each class. Each concept is linked with further words that are in one of the five different relation types with the concept: Attributive (ATTR) words describe a property of the concept. Coordinating concepts (COORD) belong to the same category. Event relations (EVENT) are verbs related to that concept. Hypernymous (HYPER) words

are in a super-ordinate relation, and meronymous (MERO) in a part-whole relation with a word. For example, see Figure 6.1, in which the concept *alligator* is in a COORD relation with *crocodile*, ATTR with *carnivorous*, EVENT with *attack*, HYPER with *animal*, and MERO with *tooth*. The set contains 14 400 word-relation pairs and 1698 unique words in these relation pairs. In addition, each concept is linked with unrelated words for distance calculation purposes. The BLESS set without the unrelated words is used in Publications V and VI.

6.1.4 Measuring distance between related word pairs

Semantic relatedness can also be directly measured instead of analyzing groups of words that form categories. A widely used test for measuring *synonymy* in the VSMs is the Test of English as a Foreign Language (TOEFL) (Landauer and Dumais, 1997). The test contains target words and multiple choices, where there is a correct answer among distractor words. Both in the human version and the VSM evaluation version, the task is to find the synonym, which in the VSM case should be closest to the target word vector in the vector space. The set can be criticized, though. Some of the words used are very rare, and hence the word vector representations for those words might be unreliable. Baroni and Lenci (2011) criticize the TOEFL test for concentrating on synonymy, which is a fairly rare semantic relation and one of the hardest to define. They also point out that it is not known how the distractor words are chosen, and the choices seem erratic: sometimes the distractors are semantically related to the target word, and sometimes not.

Other similar test sets such as English as a Second Language (ESL) and SAT college entrance exam (Turney, 2001, 2005), are also available. In addition, one can easily construct similar multiple-choice tests based on, for example, thesauri and random alternatives. For example, Moby thesaurus was used in Väyrynen et al. (2007).

Patel et al. (1997) introduced a test set containing semantically related word pairs such as *thunder-lightning, black-white, brother-sister*. In total, there are 400 words in the set. The distance from the target word to the related word is then compared to randomly picked words with a similar frequency in the corpus[3]. In Publication IV, this set was called the Distance set and the evaluation procedure was the following: The Precision

[3]Available online at `http://www.cs.bham.ac.uk/\~jxb/corpus.html`, accessed March 8, 2012

of the obtained words was calculated by checking how often the target word and the correct answer were closest, with a comparison to eight randomly picked words for each target word. As the comparison words are selected randomly, the comparison was repeated for 50 separate selections of random words and by averaging the precision over them.

Antonyms are words opposite in meaning to one another. In vector space models, antonyms tend to be found close to each other in the space, as they are used in a similar fashion. Deese antonym pairs (Deese, 1954), have also been used for VSM evaluation (Grefenstette, 1992), and included in the evaluation sets used in Publication IV. In Publication III, a similar list of 72 adjectives built by the authors was used.

As an additional resource not used in this dissertation, the Finkelstein word set can also be mentioned. It contains 535 pairs of words and their similarity scores perceived by human subjects (Finkelstein et al., 2002). In this set, the word pairs cover a varying degree of similarity from very similar to very dissimilar.

6.2 Corpus data

The size of the corpus matters when building distributional models. The larger the corpus, the better the results generally are, even if the quality of the corpus is inferior (Bullinaria and Levy, 2012).

The corpus used in the Publications III, IV, V and VI is built from all the documents in the English Wikipedia[4] that were over a size threshold of 2kB. The threshold was set in place to reduce the effect of empty or very short documents that contain little sentence structure or content. In pre-processing, all non-text markup was removed, the words were lowercased and punctuation was removed except for word-internal hyphens and apostrophes.

The representations used in VSM research are often very high dimensional. See, for example, Bullinaria and Levy (2012) for discussion. In Publication III, the 24 868 words that occur at least 100 times in the corpus were included as features. In Publications IV, V and VI, a considerably smaller feature space was used to reduce computational load: the 5 000 most frequent words. The co-occurrence count representations are calculated for the vocabulary of the 200 000 most frequent words, yielding

[4]The October 2008 edition no longer available at the Wikipedia dump download site `http://dumps.wikimedia.org/enwiki/`, accessed December 11, 2008.

Table 6.3. The error $Err = 1 - Pr$ for different vector space evaluation data. Recall $Re = 1$ unless otherwise specified. Without dimensionality reduction, with ICA, SVD and the SENNA embeddings. From Publication IV.

	5 000 feat	ICA50	SVD50	SENNA
Battig (Semcat)	0.22	0.31/0.19	0.32/0.19	0.25 (Re=0.98)
Syncat1	0.17	0.25	0.25	0.10
Syncat2	0.26	0.38	0.37	0.21
TOEFL	0.22 (Re=0.95)	0.38 (Re=0.95)	0.38 (Re=0.95)	0.34 (Re=0.91)
Distance	0.11	0.19	0.19	0.11
Deese	0.07	0.12	0.13	0.04

a co-occurrence matrix $\mathbf{X}_{200000\times5000}$. In different experiments, a subset of the full vector space was often used that corresponds to the test set vocabulary unless otherwise specified.

In Publication II, the Finnish-English part of the sentence-aligned Europarl corpus (Koehn, 2005) was used. In this corpus, there were 602 153 aligned regions, with 25 tokens per region for English and 20 tokens per region for Finnish on average. The 10 000 most frequent types in each language were selected for analysis in the context of 1 000 most frequent types in each language. After removal of duplicate items, a co-occurrence matrix of $\mathbf{X}_{19758\times1983}$ was created.

6.3 Evaluation: categories and word pairs

After introducing the evaluation sets and analysis, the discussion will now move onto summarizing the evaluation results of the Publications in this dissertation.

6.3.1 Validation of the Wikipedia VSM model using evaluation test sets

In Publication IV, the quality of the word vector space model used in Publications IV, V and VI was evaluated with seven different evaluation sets. The model was built as described in Section 6.2, and the PPMI weighting (Table 4.2) was used to smooth the co-occurrence counts. Two syntactic, and four of the semantic sets described earlier were used in analysis. These were the Battig set (called Semcat in Publication IV), the TOEFL set, the semantic word pair set by Patel et al. (1997) (called as Distance set Publication IV) and the Deese antonym set. The results, shown in Table 6.3, are in line with results with the same number of features in

Bullinaria and Levy (2012), verifying that the vector space model based on Wikipedia performs at a reasonable level. The Recall measure indicates the portion of the words in the test set that were part of the vocabulary of the Wikipedia experiment. The Recall is $Re = 1$, unless otherwise specified. Dimensionality reduction into $T = 50$ dimensions was performed both with SVD and ICA, where the dimensionality reduction is performed with PCA. For ICA and SVD, two different Error values are given. The larger error values $Err = 0.31$ for ICA and $Err = 0.32$ for SVD are obtained when ICA and SVD are performed for the whole vocabulary of 200 000 words. When calculated only for the 530 word vectors in the Battig set, the error decreases considerably. These results show that such a drastic dimensionality reduction lowers the performance in the evaluation tests, but the results are still reasonable. In this task, there is very little difference between the performance of ICA and SVD, which is expected due to the relation between SVD and PCA. To obtain results that would be at the same level as the results in the original dimensionality approximately 500 ICA or SVD component dimensions should be used. The results were also compared to the SENNA embeddings obtained with a neural language model (Collobert et al., 2011), for which the performance is at the state of the art level in various NLP tasks, such as Named Entity Recognition and Part of Speech tagging, but which had not, to our knowledge, been evaluated with the VSM evaluation sets before.

6.3.2 Analysis of adjectives

Publication III presents a study of adjective mappings. For the analysis, the authors created a set of 72 adjectives that form antonym pairs, similar to the Deese antonym set. The central motivation of this study was related to the fact that in earlier research, adjectives have been explored much less than nouns or verbs. The task was to check whether the antonym of the target word was found within its N nearest neighbors, with $N = [1, 2, \ldots, 10]$.

The results are visualized in Fig. 6.2 for the original dimensionality of 24 868 context words, a reduced dimensionality of 500 most frequent words as features, and further reduced into two dimensions with PCA, SOM and NeRV. Precision in the high-dimensional space exceeds 50 percent when two or more nearest neighbors are considered. The dimensionality reduction to 500 features reduces the performance by around 10 percentage points. Out of the three methods for reducing the dimen-

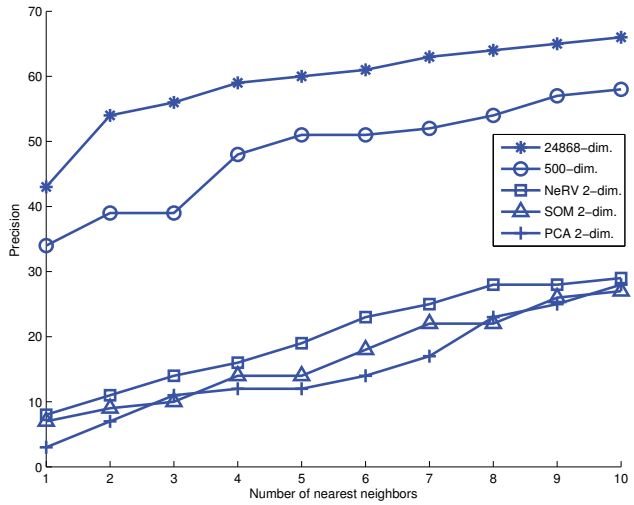


Figure 6.2. The antonym evaluation set results for Wikipedia data with different dimensionality reduction methods. On x-axis, number of nearest neighbors evaluated, on y-axis, the obtained precision: Was the antonym pair found within the nearest neighbors of the target word. Adapted from Publication III. The Recall $Re = 1$.

sionality to two which is useful for visualisation, NeRV performs slightly better than SOM, and the performance of SOM and PCA is comparable. The analysis of word similarities using visualization is useful when the number of words to analyze is high, but the downside of such a drastic dimensionality reduction is that information is lost.

6.4 Bilingual representations

The English language is presumably the most used in NLP research with a wealth of resources in the form of corpora and evaluation materials in electronic form. Most of the tools and methods used in this dissertation are unsupervised, and they can be used for other languages as well: one of the main reasons for using unsupervised methods is that languages with fewer resources can also be studied.

However, some caveats remain. When methods are developed for a particular language, such as English, different pre-processing steps might be needed to incorporate languages that convey information at a different level, for example, through a rich morphology. In addition, statistical computational methods cannot be used if there are not sufficiently large corpora in electronic form. Evaluation materials are also needed. While unsupervised methods do not require labeled material to train the model,

Table 6.4. An example of the bilingual sentence context after pre-processing and tokenization used in Publication II.

Finnish	English
eurooppalainen yhteistyö on tehotonta jos yhdenkin valtion väestö hallitus ja eduskunta pakotetaan harjoittamaan sellaista ulko ja turvallisuuspolitiikkaa jota ne eivät kannata	european cooperation would be ineffective if a country s population government and parliament were forced to conduct a foreign and security policy they did not want

some material is still needed for evaluation purposes; or in the case of exploration, an expert of the language to confirm the relevance of the findings.

ICA in a bilingual context

Publication II uses a bilingual parallel corpus, the Europarl (Koehn, 2005), to build a bilingual vector space model using a bilingual sentence context, see Table 6.4. The word co-occurrences were calculated for the words that appear in the sentence context in either of the languages. 1 000 most frequent word types were used as the features in both languages. The word vectors were built for 10 000 most frequent words in each language. Some word forms appeared in both languages, thus final co-occurrence count matrix obtained was $\mathbf{X}_{19758\times1983}$. The co-occurrence count matrix was further smoothed with a $\log(tf)$ weighting, see Table 4.2 for details. The ICA was applied to the word-context matrix built this way. The word similarity was calculated in the ICA component space $\mathbf{s} = (s_1, s_2, \cdots, s_n)$ using $T = 100$ independent components. Similarity was measured between the query word vector w_c and any other word vector using the cosine similarity, cf. Table 4.3.

The nearest neighbors were calculated for the 30 most frequent nouns both in English and Finnish, and translations of the words were searched among $N = 1, 2, 3, 4$ nearest neighbors. The translations were checked manually using a dictionary. The results are summarized in Table 6.5. On the last row of the table, the precision is reported for partial matches, i.e. when a word in Finnish would be translated into two words in English such as *jäsenvaltiot: member states*. It can be noted that the Precision is higher when Finnish query words are used, possibly because English query words can match several different inflected versions of the Finnish counterpart. Further experiments, in which stemming or lemmatization were used, might clarify this matter further.

Table 6.5. The results of the bilingual analysis of closest words of nouns. Adapted from Publication II. * Indicates precision when a part of a compound word would match. Recall is $Re = 1$.

	Query word language	
	Finnish	English
Neighbors	Precision	Precision
1	0.67	0.53
2	0.77	0.70
3	0.83	0.73
4	0.83	0.77
4*	0.96	0.83

Table 6.6. Most prominent words for sample components in the bilingual setting of Publication II. For Finnish words, translation into English is included in the basic form in angle brackets, inflection has not been marked.

saksan [Germany]	values	eroja [difference]
ranskan [France]	rauhan [peace]	different
germany	demokratian [democracy]	difference
france	vapauden [freedom]	välillä [in between]
french	democracy	erilaista [different]
german	ihmisoikeuksien [human rights]	differences
sweden	arvoja [values]	erot [difference]
netherlands	solidarity	toisiaan [each other]
ranska [France]	peace	disparities
belgian [Belgium]	arvojen [values]	eri [different]
ruotsin [Sweden]	kunnioittaminen [respect]	erilaiset [different]
saksa[Germany]	oikeusvaltion [constitutional state]	differ
italian	principles	differing
kingdom	continent	eroavat [to differ]

6.5 Finding category information

It has been shown that Independent Component Analysis can produce cognitively meaningful components (Hansen et al., 2005) that correspond to for instance noun concepts (Chagnaa et al., 2007), phonological categories (Calderone, 2009), personal traits (Chung and Pennebaker, 2008) and syntactic categories (Honkela et al., 2010). In Publication II, we identified components that contained semantically related words in a bilingual setting. Examples are shown in Table 6.6, with English translation for the Finnish words. In that work, it was shown that ICA finds semantically related word sets, but no further evaluation was carried out. Publication IV begins a series of experiments to find out what kind of semantic representations and category information can be obtained with unsupervised methodology such as ICA. The research direction is continued in Publi-

cation V with a second evaluation set and Publication VI where ICA is compared to a method based on Latent Dirichlet Allocation.

In vector space related research, several different terms have been used to describe semantically related words or semantic categories, such as 'emergent category' (Honkela, 1998), 'latent class' (Hofmann, 1999), 'probabilistic word class' (Chrupała, 2011), 'topic' (Blei et al., 2003; Steyvers and Griffiths, 2007) and 'sense' (Brody and Lapata, 2009). Out of these, the four first ones are often used synonymously, whereas the term 'sense' is often used when multiple meanings of words are considered. Essentially, the phenomenon is still the same: what kind of semantic distinctions are made?

The task of finding category information is related to the task of sense induction. See, for example, Brody and Lapata (2009). In finding category information, we try to group words into meaningful clusters, whereas in sense induction the task is to automatically identify different word senses from the corpora. This in turn differs from the traditional task of word sense disambiguation, where the senses are assumed to be known and fixed. The LDA based methods for word sense induction have been recently developed (Brody and Lapata, 2009; Dinu and Lapata, 2010; Chrupała, 2011), but Independent Component Analysis has also been used for this task (Rapp, 2004).

6.5.1 Comparison to evaluation sets

In Publication IV, a method to analyze the category information possibly represented in the components was devised, first for the Battig data set. For this purpose, the maximum activation of the components or topics was studied. In the case of ICA, the components are usually skewed in one direction, and the analysis was done in the direction of maximum skewness. For each component or topic, the words were sorted in the order of the value of the activation, and N words with the highest value were chosen for analysis. In the experiments, $N = 10$ was used which corresponds to the number of words in each category in the Battig set. The limit is somewhat arbitrary, and could be also set based on, for example, a certain activation threshold.

Two different Precision thresholds, *strict* and *lax* were defined following Sahlgren (2006). In the strict case, a minimum of $Pr_{strict} = \frac{9}{10}$ words belong to the same category and in the lax case, a minimum of $Pr_{lax} = \frac{6}{10}$ words belong to the same category. The analysis is used in three studies,

Table 6.7. The fraction of Battig categories found with strict and lax condition analyzing the component information for ICA and SVD and the SENNA embeddings, from Publication IV

	Strict	Lax
subset+ICA	17/53	37/53
ICA+subset	1/53	12/53
subset+SVD	2/53	19/53
SVD+subset	3/53	8/53
SENNA	0/52	4/52

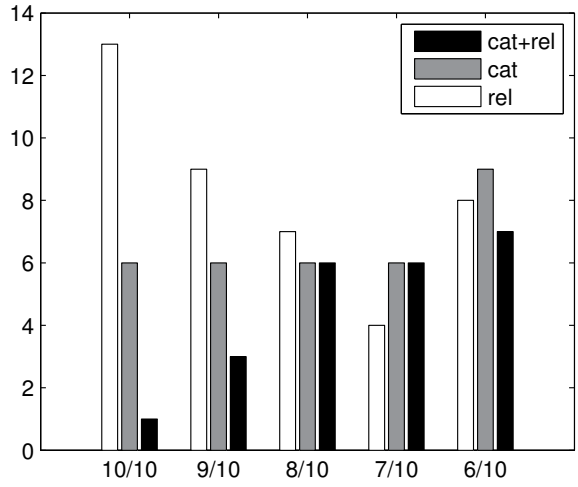


Figure 6.3. The results for the BLESS categories and relations for ICA for different thresholds. On x-axis, the different analysis thresholds used and on y-axis, the number of categories or relations found. From Publication V

with different comparison methodology and test sets.

Comparison of ICA and SVD with Battig set

In Publication IV, the performance of ICA and SVD was compared using the Battig set of 53 categories described earlier using a reduced dimensionality of $T = 50$ components. The selection of the subset of word vectors can be carried out either before or after the dimensionality reduction. It is obvious that if the method is applied after the subset selection (subset+ICA/SVD), the components are a better representation of only those words in the subset, than if the subset selection is carried out after a dimension reduction for the complete matrix of 200 000 word vectors (ICA/SVD+subset). The results in Table 6.7 indicate that ICA performs better than SVD in the task of extracting interpretable components, that is, finding groups of words that correspond to a semantic category.

ICA experiments with the BLESS set

The words in the Battig data set are all nouns. In Publication V, the BLESS data set introduced in Section 6.1.3 with a larger vocabulary and richer labels was used. Out of the unique words in the BLESS set, 1673 words were found in the vocabulary of 200 000 words of the Wikipedia data used. The analysis was carried out separately for the 17 categories, 5 relation classes, and 85 joint category-relation combinations (e.g, WATER_ANIMAL-MERO, VEHICLE-COORD). Multiple labels for each word were also allowed. For example, the word *tooth* from Fig. 6.1 has four category labels as it appears in relation to a word in four different categories: AMPHIBIAN_REPTILE, GROUND_MAMMAL, TOOL and WATER_ANIMAL. It appears in all of these categories in a meronymous relation, thus the only relation label is MERO. It also has four joint category-relation classes (WATER_ANIMAL-MERO, etc...) created by combining the category and the relation label for more fine grained analysis.

The experiments were run on the subset of the BLESS words using $T = 50$ independent components, again analyzing the 10 words with highest activation for each component. The evaluations were carried out on additional thresholds of $Pr = \frac{7}{10}$ and $Pr = \frac{8}{10}$. The results are given in Figure 6.3. This study confirmed that the words with maximal value for a given component often belong to the same category or relation defined by the BLESS labels, although the components do not correspond to the fine grained cat-rel labels very often: only 23 out of 85 cat-rel groups are covered with the lax condition. On the other hand, 12 out of 17 categories are covered at the lax threshold using 33/50 components, and 41/50 components cover the five relation types well. The different category and relation types are analyzed in more detail in the following.

Comparison of ICA and LDA using both Battig and BLESS sets

Probabilistic topic modeling, for example, using Latent Dirichlet Allocation (LDA), is a prominent approach created especially for text data. The performance of the ICA and LDA-based method was compared in Publication VI, with different model sizes or number of topics or independent components T, using both the Battig and BLESS data sets and 10 iteration runs for each model size. The ICA model is trained with the Battig or BLESS vocabulary using the PPMI weighting (Table 4.2). In LDA experiments, weighting is not usually used (Wilson and Chew, 2010), but, for example, Dinu and Lapata (2010) use a simple scaling of counts by a factor

Table 6.8. Number of Battig categories found: results for strict (S) and lax (L) criterion for different model sizes for ICA, LDA 1 (ppmi-ceil), LDA 2 (ppmi-ceil long) and LDA 3 (noweight). The highest value for each model size and condition is marked in boldface. From Publication VI.

		Model size									
	Model type	10	20	30	40	50	60	80	100	150	200
L	ICA	**7.8**	**14.1**	**23.0**	**34.0**	**39.3**	36.3	37.4	39.7	39.1	34.2
	LDA 1	4.8	12.0	22.7	28.9	36.2	38.1	41.3	**43.1**	49.0	52.3
	LDA 2	4.1	11.9	21.5	29.6	36.4	**39.7**	**42.9**	43.0	49.0	53.1
	LDA 3	4.1	12.3	21.8	30.9	36.0	38.9	41.8	**43.1**	**51.4**	**54.1**
S	ICA	**1.0**	**8.1**	**11.0**	**16.3**	**17.0**	16.2	17.2	15.1	**14.8**	9.0
	LDA 1	0.0	1.2	3.9	10.3	14.8	16.3	18.4	16.2	13.8	**10.7**
	LDA 2	0.0	1.2	3.6	10.8	14.8	17.8	**19.9**	**16.8**	12.8	8.9
	LDA 3	0.0	1.4	3.7	9.7	14.0	**18.7**	18.8	16.5	11.8	10.4

of 70. In Publication VI, the LDA-based model was tested with two different weighting alternatives. The first alternative was a PPMI-weighting based heuristic, where all the PPMI-weights of the matrix were rounded up to nearest integer. It was tested with a shorter training length of 500 iteration steps (LDA 1) and a longer training of 2 000 iteration steps (LDA 2). The second alternative was no weighting at all, computed with 500 iteration steps (LDA 3). The long training was not implemented in the case of no weighting as the calculation times grew very long. See Publication VI for further details and comparisons.

The results for the analysis for the Battig set are given in Table 6.8, averaged over 10 iteration runs. With a smaller model size, ICA finds more categories than the LDA model variants but as the model size approaches the number of the categories in the Battig set, the performance difference evens out. When the model size is approximately the same as the number of categories in the Battig set, $T = 50$, the ICA model is able to find 39 categories with the lax condition and 17 categories with the strict condition. The results of the LDA variants are slightly worse. The models find approximately 36 categories with the lax condition, and 14 categories with the strict condition.

When the model size grows larger than the number of the Battig categories, performance of ICA declines as the categories are split into several components. The LDA does not suffer from this behavior but continues to use several topics to describe each category, as the current model does not penalize for using several topics or independent components to represent a single category. The results with the BLESS set were similar, see Publication VI for details. Thus it can be concluded that both models are able to find a considerable number of semantic categories in an unsupervised manner.

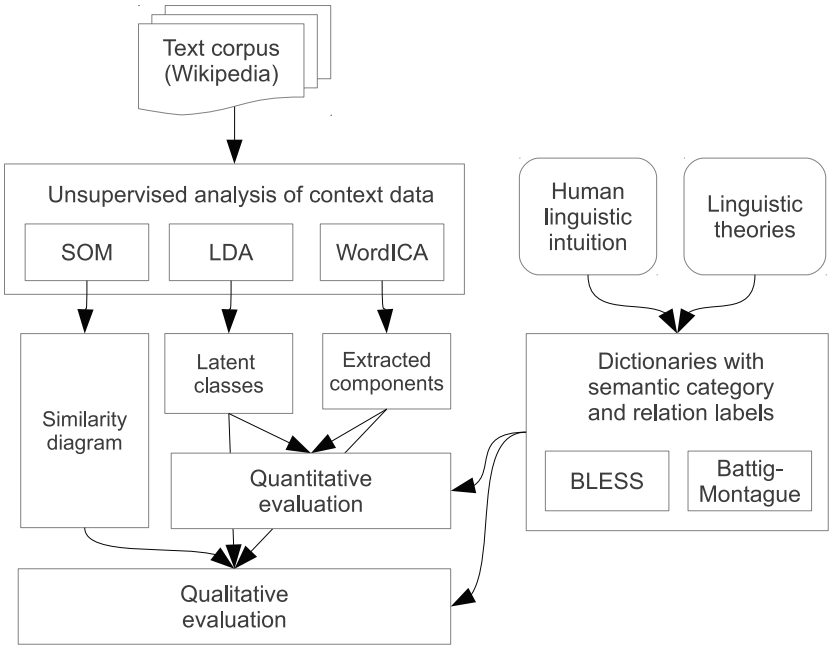


Figure 6.4. The unsupervised methods for text analysis as an explorative tool. From Publication VI.

6.6 Exploration

Corpus-based approaches can provide a way to take into account cognitive processes that limit the actual applicability of a certain linguistic rule. See, for example, Karlsson (2007). In similar vein, an explorative approach may be useful when considering lexical semantics. Human linguistic intuition and theories provide dictionaries with semantic labels, that are used to evaluate the results obtained with the unsupervised methods. The theories, and especially the similarity judgments, are in reality subjective and depend on the context. In addition, the coverage of the semantic labelings is often limited. Publications V and VI discuss the use of the unsupervised learning methods as an explorative tool. The schematic description of the process is shown in Fig. 6.4. The purpose of the work was both to check how well the representations generated in a data-driven manner coincide with the manually constructed semantic categories; and to analyze the qualities of the manually constructed semantic categories using statistical machine learning and visualization.

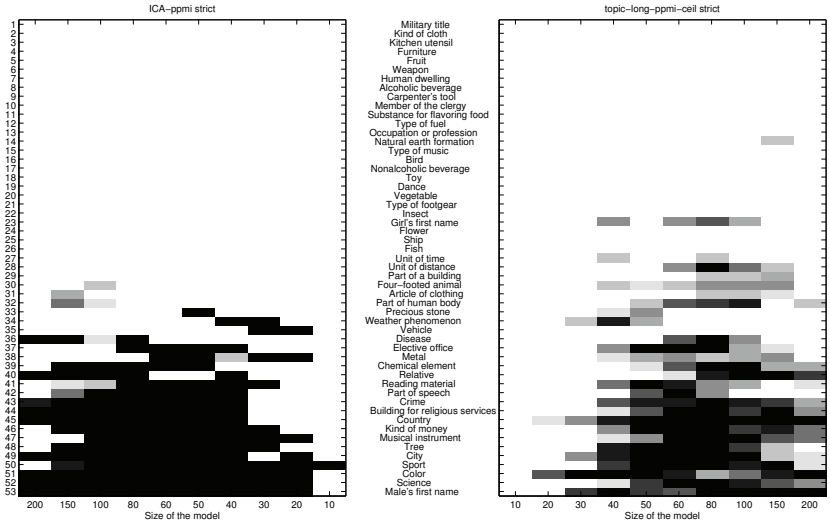


Figure 6.5. Comparison of the ICA (left) and LDA 2 (right) results for the Battig categories using the strict criterion. The model size is on the x axis, and the categories on y axis. The categories found best are at the bottom, those not found at top. From Publication VI.

6.6.1 Good and bad categories: an analysis of the evaluation sets

Publication IV reports on an experiment in which forward feature selection based on entropy was used as a means to separate one of the Battig categories from all of the others, see Figure 4 in Publication IV. The performance in feature selection was further compared to a separation of randomly generated categories. It was found out that while the average separation of the Battig categories was clearly better than random, there were some categories that were not separated well.

In Publication VI, a visualization method was devised to show how well the categories are found for different model sizes. Figure 6.5 shows the results for the strict condition using the ICA and the LDA 2 model and 10 iteration runs. The darker the rectangle on the visualization, the more often it was found on different iteration runs, and a white rectangle indicates that the category was not found. The resulting visualization of the LDA-based model contains lighter shades of gray, probably due to the random initialization. On the other hand, in ICA, the Principal Component Analysis step is always the same, and variation between the iteration steps is smaller. It can be seen that some categories are found early on with the strict condition, such as COLOR, MALE NAME, VEHICLE, SCIENCE and NATURAL EARTH FORMATION. The LDA model finds more categories with the strict condition, but the categories are not necessarily found on

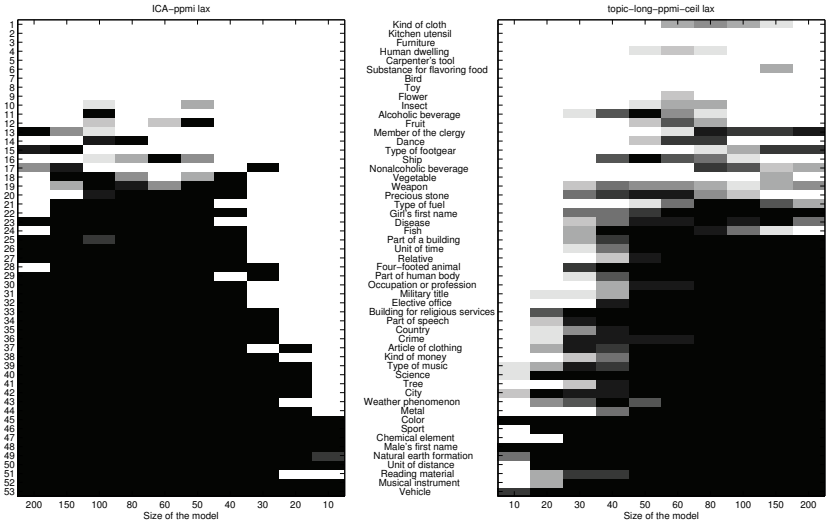


Figure 6.6. Comparison of ICA (left) and LDA 2 (right) results for the Battig categories with the lax criterion. From Publication VI.

every iteration. The effect of model size is also evident: When the model size grows larger than the number of categories in the set, words of the category are split into different topics or components, and the category is 'lost' again.

Figure 6.6 shows the results for the Battig data for the lax condition. A majority of the categories is found with this condition. Interestingly, some categories, such as KIND OF CLOTH, KITCHEN UTENSIL, FURNITURE, CARPENTER'S TOOL, and TOY are rarely found with the LDA model, and the ICA model does not find them at all. In Publication VI, similar analysis was also made for the BLESS categories and relations with similar results on the BLESS categories (see Figures 5, 6, and 7 in Publication VI).

These results suggest that while it is easy for a human to make category judgments of almost anything, there are quality differences between the categories in these sets which makes some of them more difficult to be represented with a vector space model. This phenomenon is further studied in the following section.

6.6.2 Visualizing words and category relations

Visualization of the words in a map that preserves maximally the neighborhood structure can also be a useful tool to analyze the relations between the words. There is a long history of visualizing word maps, often with the SOM (Kohonen, 1982, 2001), starting from Ritter and Kohonen

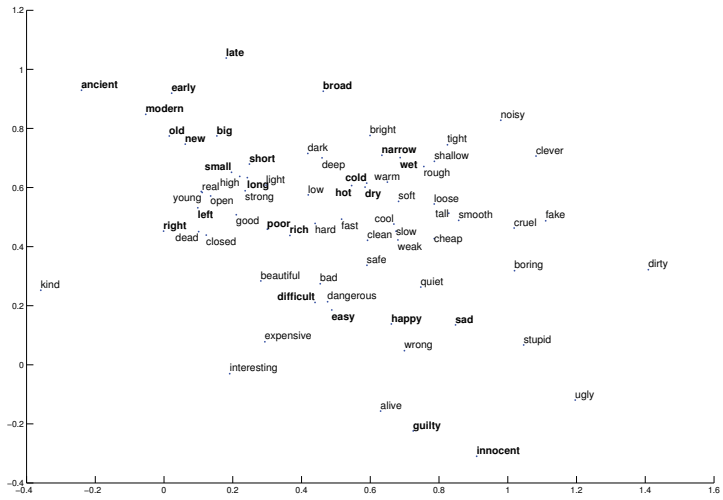


Figure 6.7. The antonym pairs from Publication III visualized with NeRV. The antonyms which have their pair in the close neighborhood are shown in boldface.

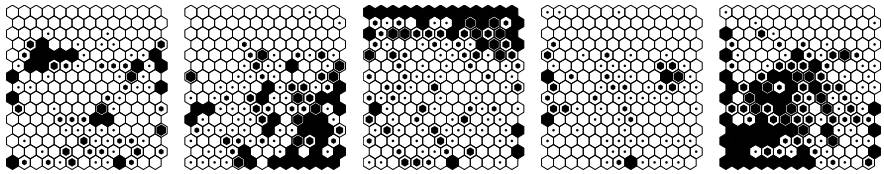
(1989) and Honkela et al. (1995), which concentrated more on syntactic categories. The SOM is also useful when visualizing document collections (Kaski et al., 1998; Back et al., 2001). In this dissertation, adjectives were visualized with NeRV in Publication III and the BLESS and Battig categories were visualized with the SOM in Publications V and VI.

6.6.3 Visualization of adjectives with NeRV

Figure 6.7 shows a visualization from Publication III, where the antonym pairs are projected into a 2-dimensional space using NeRV. The antonyms that have their pair in the close neighborhood are shown in boldface. This visualization shows that some similar words are close to each other, such as the time-related adjectives at the top left corner. Such a visualization can be used as a tool to inspect relations of words. In the case of the adjectives, there are some word pairs and semantically related words that are close by, but no clear division.

6.6.4 Visualization with SOM hit histograms

After training a SOM with the word vectors, the words can be visualized on the Self-Organizing Map by adding labels to map units that best match the word vectors. This approach is suitable, if the number of labels is limited. Too many labels make the visualization illegible. Categories can be also visualized with hit histograms. The data points are projected to the map to the corresponding best-matching units, and the size of the dot in



(a) Attribute **(b)** Coordinating concept **(c)** Event **(d)** Hypernym **(e)** Meronym

Figure 6.8. The different relation classes of the BLESS set visualized with the Self-Organizing Map. A good separation has been obtained for classes that do not overlap with other categories, such as the event (c). From Publication V.

each map unit indicates the number of hits. In the following visualizations, a completely filled map unit contains five or more hits. In Publication V, the SOM was trained with the word vectors that correspond to the BLESS vocabulary, and in Publication VI a joint vocabulary that contains both BLESS and Battig vocabularies was used. The label information is only used in visualization, and it does not affect the training of the map.

Visualization of BLESS relation classes with SOM

In Publication V, the BLESS relation classes were visualized on a Self-Organizing Map, shown in Fig. 6.8. The words in the ATTR category are all adjectives, the words in the EVENT category are verbs, and the words in the COORD, MERO and HYPER categories are nouns. It can be seen that the EVENT category Fig. 6.8c separates well from the others. The ATTR class (Fig. 6.8a) separates into several disjoint regions, and the HYPER class is similarly spread into several disjoint regions, some overlapping the ATTR and MERO classes. The two other noun categories, MERO and COORD, are fairly separate.

Visualization of Battig categories with SOM

In Publications V and VI, the categories that were not found with ICA or LDA were analyzed in more detail using the SOM hit histogram visualizations. In Figure 6.9, ten categories from the Battig set are visualized. Figs. 6.9a–6.9e show the five categories that were found best with LDA and ICA using the strict condition and Figs. 6.9f–6.9j the five most difficult categories that were not found with LDA or ICA using the lax condition.

The visualizations show that the 'easy' categories form a concise cluster of one or few neighboring map units, whereas the difficult categories are often spread all over the map without forming a concise cluster. The only 'easy' category with an outlier in these visualizations is the category

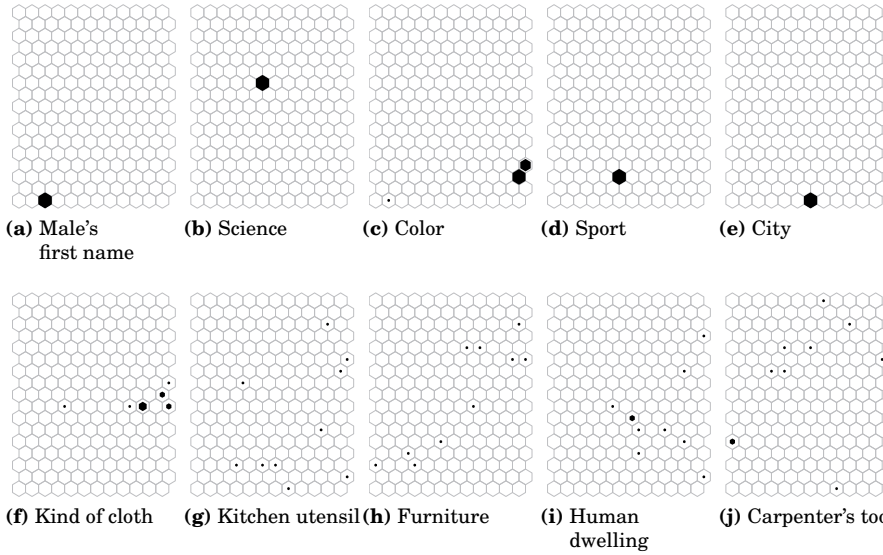


Figure 6.9. The best (a–e) and worst (f–j) Battig categories visualized with the SOM. Adapted from Publication VI.

COLOR in Fig. 6.9c. The outlier in the lower left corner is the word *brown*, which is mapped to a node where the other instances belong to human categories, indicating the polysemous nature of the word. This is often the case with the words in the 'difficult categories'. The words in them seem to be more often common and polysemous, whereas in some cases, such as KIND OF CLOTH in Fig. 6.9f, there is also overlap with another other, more prominent category, ARTICLE OF CLOTHING.

These results make us pose a question: What kind of features would be needed to represent these 'difficult' categories? Would a representation that distinguished between different context types, proposed, for example, by Erk and Padó (2008) be sufficient? The BLESS category visualizations with similar results can be found in Figure 5 of Publication V.

Related categories

The relations between categories can be examined using a SOM visualization. In Publication VI, the words in the seven Battig categories that belong to a higher-level category HUMAN were visualized using a hit histogram and adding labels to denote the hits in each map unit. The results are shown in Fig. 6.10. The categories overlap partly and most of the words are mapped together in the lower left corner of the map except for two outliers. These correspond to the words *private* and *major* from the MILITARY TITLE category and they are close to attribute words (see Fig.

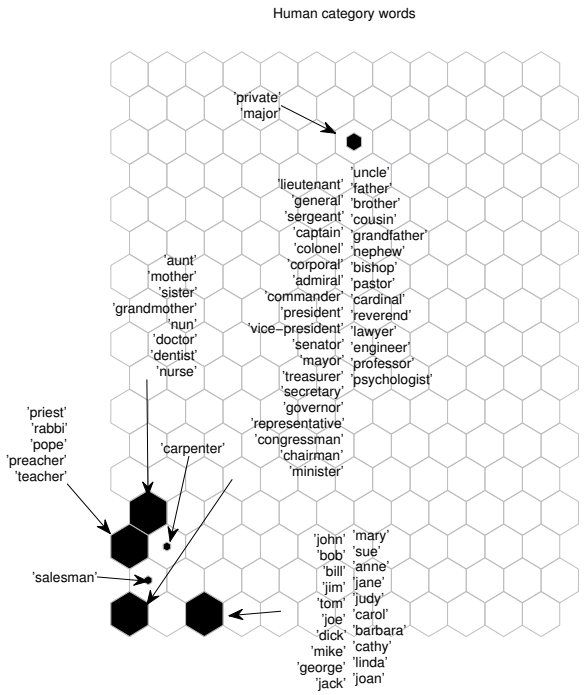


Figure 6.10. All the words of the categories in the Battig set that are part of a higher level category HUMAN. These categories are MALE NAME, GIRL'S NAME, MEMBER OF THE CLERGY,ELECTIVE OFFICE , RELATIVE, OCCUPATION OR PROFESSION and MILITARY POSITION. From Publication VI

.

9a in Publication VI), which indicates that the sense induced from the distributional data is not the one they were labeled with.

A split between the genders can be also noted: While the map unit at the lower left corner is the most populated one containing all words from the ELECTIVE OFFICE category, and most words from the MILITARY TITLE, it also contains the male words from RELATIVE. The member of the clergy category seems to be divided into two main parts. *Bishop*, *pastor*, *cardinal* and *reverend* are also in this map unit, whereas words such as *priest*, *rabbi*, *preacher* and *pope* are in another map unit along with *teacher*. This might indicate similarity in the meaning of words *preacher* and *teacher*, but no obvious reason that would cover all of the words mapped in this unit can be given. In a separate map unit, there are words indicating female gender, including the female words from RELATIVE and words indicating female occupation such as *nun* and *nurse*. In addition, the words *doctor* and *dentist* are mapped onto this node. The proper nouns are all mapped into a separate map unit regardless of the gender. This kind of an analysis can benefit exploration by making the relations between cat-

Table 6.9. Words with highest activation for a sample of ICA components using 50 independent components. The characterizations on the first row are given by the researchers. From Publication V.

'moving'	'material'	'music'	'attribute'	'fruit'	'color'	'cultural'	'animal'	'tree'
dive	steel	rock	male	strawberry	red	christian	insect	pine
jump	wooden	pop	healthy	pineapple	blue	medieval	mammal	cedar
crawl	ceramic	music	young	banana	yellow	indian	vertebrate	oak
swim	plastic	dance	solitary	citrus	white	ancient	invertebrate	cypress
kick	metal	acoustic	female	mango	purple	modern	aquatic	poplar
glide	concrete	hop	intelligent	grape	black	american	carnivorous	willow
walk	glass	jam	timid	peach	pink	religious	reptile	evergreen
climb	cardboard	metal	shy	apricot	green	asian	amphibian	birch
fly	copper	mix	faithful	watermelon	grey	african	bird	elm
float	iron	swing	peaceful	lemon	golden	roman	animal	acacia

Table 6.10. The number of frequent sets per each set size analyzed for strict (S) and lax (L) condition. ICA is able to find more stable frequent sets than LDA. From Publication VI.

			Set size				
Model size	Condition	Method	10	9	8	7	Total
60	S	ICA	8	10	12	9	39
		LDA 1	0	3	7	2	12
		LDA 2	0	3	3	7	13
	L	ICA	13	27	23	18	82
		LDA 1	1	12	16	12	41
		LDA 2	3	14	15	18	50
200	S	ICA	9	20	22	23	74
		LDA 1	0	2	5	4	11
		LDA 2	0	2	7	12	21
	L	ICA	32	36	38	43	149
		LDA 1	1	12	25	31	69
		LDA 2	2	15	29	52	98

egories visible, show possible subcategories (such as gender), and polysemous words which are not clustered according to their 'indicated' category label.

6.6.5 Qualitative analysis of frequent related word sets

Independent Component Analysis can be used to find interesting word sets in an explorative manner. In Publication V, several sets were shown to be of interest, listed in Table 6.9. They range from verbs describing movement to music-related words, materials, fruits and cultural attributes.

Publication VI introduces a method to find sets of semantically related words that occur frequently in separate iteration runs using a simple search algorithm. Ten iterations were run in each case of the analysis. First, the word sets of size $N = 10$ that correspond to the maximum activation for each component or topic are selected. As it was noticed that there is sometimes slight variation of one or two words between iteration

Table 6.11. Types of qualitative classes the word sets found were classified into

Type	Description of the qualitative class
A	Words are related in some way and the majority label given is as descriptive as possible of the words in the set.
B	Words are related in some way and the majority label is somewhat descriptive, but a more descriptive account can be easily given.
C	Words are related in some way, but there is no majority label that describes the words
D	There is no majority label, nor is there any perceived relation between the words in the set.

runs, the subsets of the original sets are taken into analysis, defining the size of set as $10 \geq N_{set} \geq 7$. From all possible word sets found, the sets that occurred in different iteration runs more frequently than a threshold M were retained for analysis. The strict and lax limits were again defined: In the strict condition, the word set had to be found on nine out of ten iteration runs, $M_{strict} =\geq \frac{9}{10}$, and in the lax condition, it had to be found in a majority of the iteration runs, $M_{lax} \geq \frac{6}{10}$. Note that earlier Pr_{strict} and Pr_{lax} were defined as the number of words in a set that belong to a certain evaluation category. In the current experiment, there is no evaluation set: the M_{strict} and M_{lax} refer the times the same word set is found over separate iteration runs.

Results were reported for model sizes $T = [60, 200]$ for both strict (S) and lax (L) condition, with models trained on the word vectors of the BLESS vocabulary. Table 6.10 details the number of sets found for each model type, size and condition. The ICA method is able to find a considerably larger number of frequent sets with both conditions than the LDA method. The training length of the LDA method affects also the number of frequent sets. With longer training (LDA 2), there are more stable frequent sets that are found in different iterations. This phenomenon is especially clear with the large model size (200).

Further, an analysis on the *quality* of the sets found was carried out in Publication VI. In exploration, there might not be labeling available for all semantic groupings, and human evaluators must be used instead. To obtain insight on the quality of the retrieved word set, a simple evaluation criteria was devised. Each word set exceeding the criteria described above was checked against all existing BLESS labels, and majority labels for category, relation and category-relation were calculated for each word

Table 6.12. Examples of the word sets of different qualitative types taken from an analysis of ICA with 200 components and the strict condition. From Publication VI.

A	B	C	D
cannon	aeroplane	dance	circuit
cartridge	aircraft	electronic	floor
firearm	airplane	garage	fret
grenade	bomber	hop	pick
gun	fighter	jam	place
musket	glider	metal	round
pistol	helicopter	music	season
revolver	jet	pop	seed
rifle	pilot	rock	spot
shotgun	plane	swing	ward

set Four different qualitative classes or types were then defined. These types characterize the quality of the set: Are the words semantically related, is there a majority BLESS label, and does it describe the group well? The types are listed in Table 6.11. These types are A, a descriptive majority label; B, a somewhat descriptive majority label but more specific description is easily found; C, no descriptive majority label exists, but words are clearly related; and D, no descriptive majority label, nor any clear semantic relation between words is easily given.

Each word set was classified into one of the four types based on how well the well the majority category or relation labels described the word set. These classifications are given below. Examples of the different categories are given in Table 6.12. In the example of type A, all words are correctly labeled with WEAPON. In the second example, of type B, all words are labeled with VEHICLE. This as such is correct, but in addition, all words except *pilot* are vehicles that fly. The methods are able to represent land vehicles and water vehicles with different components. In the third example (type C), there is no category label to describe the majority of the words, but an evaluator can easily see that the words are all related to music. Finally, in the fourth example (type D), there was no easily perceived semantic sense for the group by the researchers.

The word sets were then analyzed by the researchers. The number of types found with each model type and size are shown on Table 6.13. The existing BLESS labels (type A) coverage varies between 0 and 40 percent, whereas the case B coverage is between 20 percent and 80 percent. This is mostly due to the word sets labeled with a simple relation label divided into more distinguished sets. The fraction of the meaningful sets with no label (type C) is not negligible, either. The coverage ranges from seven to

Table 6.13. The qualitative analysis of the found frequent sets in the BLESS set. From Publication VI.

Model size	Condition	Method	A	B	C	D	total
60	S	ICA	8	22	6	3	39
		LDA 1	5	3	1	3	12
		LDA 2	5	4	2	2	13
	L	ICA	16	50	8	7	81
		LDA 1	13	16	8	4	41
		LDA 2	12	33	3	2	50
200	S	ICA	7	49	9	9	74
		LDA 1	0	9	1	1	11
		LDA 2	0	17	2	2	22
	L	ICA	23	101	11	14	149
		LDA 1	8	48	5	7	68
		LDA 2	13	60	17	8	98

20 percent of the sets found, depending on the number of topics and the condition.

In Publication VI, the analysis is carried out only by the researchers. Thus it is subject to the same subjectivity as any labeling, and the results give only a rough estimate about the quality of the word sets discovered. It can be noted that for majority of cases, the existing labels give some kind of a description of the words, and partial labels, which, for example, describe the words as attributes, are more prominent. The number of word sets found without a descriptive label was relevant, as well, and the number of nonsense word sets was fairly low.

Examples of semantically meaningful word sets found with ICA (with $T = 200$ components) are shown in Table 6.14, with a characterizing description given on the first row. Very different attribute groups from cultural attributes, to colors, and attributes related to sensing were discovered. Characterizing noun categories is easier than characterizing attribute or adjective classes, but these results demonstrate that such groupings can be found from corpus data with distributional analysis and unsupervised methods.

Table 6.14. Different word sets labeled with the attribute relation from the ICA-200 set with strict condition. The characterizations have been given by the researchers. From Publication VI.

'cultural'	'time/period'	'positive/ negative'	'dangerous'	'short'
african	ancient	accurate	aggressive	bad
american	antique	bad	armed	cute
ancient	baroque	excellent	bitter	dirty
antarctic	gothic	fresh	ferocious	funny
asian	medieval	good	fierce	nice
christian	modern	impressive	heavy	pretty
indian	old	magnificent	sharp	scary
national	roman	solid	strong	stupid
rim	romanesque	strong	stubborn	ugly
roman	ruined			

'color'	'shape'	'taste'	'dangerous'	'animal characteristics'
black	circular	crunchy	addictive	aquatic
blue	curved	delicious	dangerous	arboreal
green	cylindrical	juicy	deadly	carnivorous
grey	flat	oily	destructive	endangered
pink	narrow	sour	explosive	herbivorous
purple	oval	spicy	lethal	nocturnal
red	rectangular	sweet	nuclear	solitary
white	rounded	tart	poisonous	venomous
yellow	spiral	tasty		

7. A simulation model of concept and lexicon emergence

This chapter details a multi-agent simulation approach for modeling vocabulary emergence presented in Publications VII, VIII and IX. In this chapter, the focus is on the process of an emerging language, categorization and naming. The model introduced in this chapter is a complete model in a sense that the language used is grounded, and the three corners of the semiotic triangle introduced in Section 2.2 are included. This, according to Vogt (2002), suffices to solve the symbol grounding problem. In this approach, the model is borrowed from Peirce, but does not go deep into Semiotics. (See, for example, Gomes et al. (2007) for discussion on computer simulation of Peircean signs in a strict sense.) This approach can be contrasted to the distributional model presented in the previous chapter, where the meaning of words is grounded only through use. Compared to the natural language discussed in the previous chapters, the language used in the simulations is artificial, but emergent similarly to natural languages. Refer to Table 2.1 in Chapter 2, in which different types of languages were defined.

The goal of the work presented in this chapter is to study how a shared vocabulary *emerges* in the interaction in a *population of individuals*. For this purpose, modeling choices for both a) the model for an individual and b) for modeling the communication between individuals are discussed. Table 7.1 lists minimal abilities the agent needs for the communication purposes. The perception ability allows the agent to receive information from the outside world. To perceive, the agent needs senses to receive the input through the senses. The conceptual representation ability requires a conceptual memory in which to store the concepts, and a method by which to produce the concepts from the perceptions. The theoretical discussion of the concept of concept was presented in Chapter 2. The ability to communicate using symbols assumes several other skills: The agent must, at

Table 7.1. The semiotic triangle and the abilities of the agent required

Semiotic triangle	Abilities of the agent	Computational realization
Referent	Has senses to receive sensory information	Can perceive objects described by a three dimensional (color) vector
Concept	Can produce conceptual representations from perceptions and store and access and recognize them	A conceptual memory realized by a SOM with a possibility to map perceived objects and words into the map
Symbol	Ability to produce words	Creates words from a given 'language'
	Ability to perceive the words signaled by another agent	Can 'perceive' words uttered
	Ability to associate the symbols to the concepts	Can associate words to the best-matching unit of the viewed object
		Ability to select best word for a given utterance based on selection criteria

least, be able to perceive and to produce symbols, and associate the symbols to the concepts; have a learning mechanism to be able to decide which symbol to use; be able to establish joint attention; and be able to engage in a communication act.

Figure 7.1 illustrates the two modeling processes at the level of the community. Each agent is an individual with its own conceptual model and subjective experiences that train the conceptual memory. The agents engage in communication acts, and through these interactions a shared vocabulary emergence can be studied. By a shared vocabulary, it is meant that statistically speaking, two agents use the same symbol to describe the same referent. See Section 7.2 for details. This chapter details the computational realization choices for such a model, the experiments and the evaluation of the results.

7.1 From perceptions to concepts

In the following, the modeling assumptions and choices are discussed in more detail.

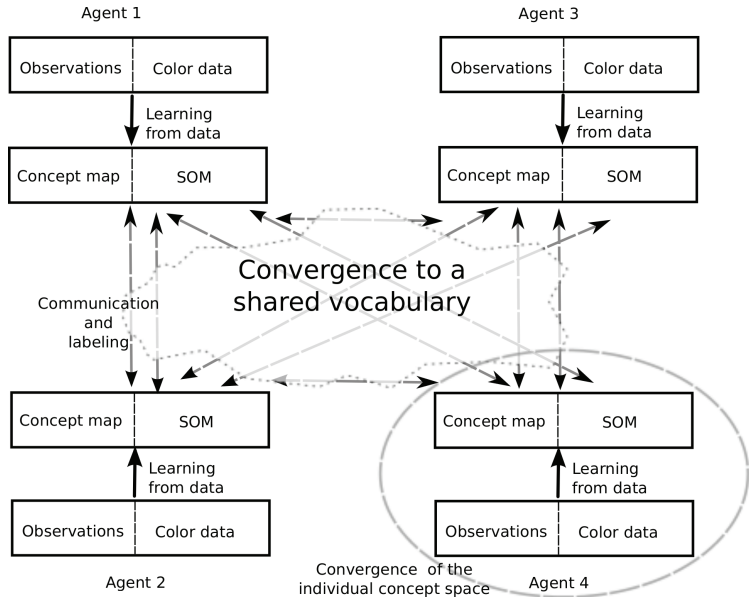


Figure 7.1. The two processes in the simulation model: Learning the individual concept space for each agent separately, and convergence to a shared vocabulary using communication instances between two agents in a population.

Subjective experiences

We assume that the observations of the agents are *subjective*, while the underlying machinery to treat them is the same. This corresponds to the case where the biological capabilities of the humans are approximately the same, but the experiences during learning affect the representation level: the formation of the concepts.

The sensory information

In Publications VII and VIII, the agents perceptions consist of colors represented in the Red-Green-Blue (RGB) color space $\mathcal{C}$, which is a three-dimensional continuous space with $c_i \in [0, 1]$. The data source are artificial color pictures, see Fig. 7.2 for an example. The data has 8 peaks or prototypical colors $\boldsymbol{\rho} = \{000, 001, 010, 011, 100, 101, 110, 111\}$ corresponding to the eight corners of the 3-dimensional cube that makes up the RGB color space, and noise at the level of 20 percent is added to make the peaks wider. Thus, the perceptions of the agents are continuous, and not only limited to the eight discrete prototypical colors. The amount of each color in each picture varies. Any other color representation could have been used in the place of the RGB color space.

Separate training data sets for training the semantic memories of each agent were used in the experiments. This mimics the situation where

Figure 7.2. An example of the color picture used to train the semantic memory of an agent. Each agent is trained with a slightly different picture containing the same prototypical colors and noise at the level of 20 percent.

the experiences of the agents differ. For the communication phase, an additional data set was constructed in a similar fashion to serve as the topics of the language games that the agents play. The topic for each communication instance was then selected randomly from this set.

7.1.1 The process of concept learning

In the model, concept learning is divided into two phases. First, the individual concept maps are trained with the color data prior to the language acquisition. After the initial training, the map is not changed. This corresponds to a situation, in which a child would initialize its feature representation based on natural visual data, prior to any vocabulary learning. This approach is a simplification, and in a realistic setting the concept space and vocabulary co-develop at least partly. In the first phase, the agent learns the regularities in the data. There are no fixed boundaries between concepts, but the concept representations are more or less continuous. See, for example, Zadeh (1965) and Honkela and Vepsäläinen (1991) for further discussion. This differs from the approach by Vogt (2005), for example, in which a separate discrimination game was used. In that game, the agent explicitly partitioned the space in such a way that each item in the visual view of the agent was in a separate partition in the space that formed a concept.

In the second phase, when the agents start playing the communication games, words are associated with the concepts in the concept space. This is realized by mapping the observation to the closest match in the semantic memory, and attaching the name as a label to that map unit.

The agents create new words if there are no words they can use. Each map unit may be associated with multiple labels, and labels can be associated with multiple map units, hence the mapping is many-to-many. As there are no explicit boundaries, proximity is used as a measure of similarity, and names can be shared with items mapped to nearby units in the semantic memory.

It is clear that the two-phase model does not fully correspond to human word learning, where the association between words and the referents is an ongoing process and parallel rather than serial. In human children, also language comprehension precedes language production, which allows humans to transfer the existing language to the young. Such a model would be easy to realize as a generational model, such as the Iterated Learning Model (ILM) (Kirby, 2001), in which only the older, adult agents can be speakers, and children are always the hearers.

An agent model also includes word production and word perception. In the current model both are very simple. Words are perceived as transmitted, and possibility of error in the communication channel is not considered. Word production is also assumed error free. This is in contrast with the noisy channel model (Section 3.4.2), where the focus is in the effect of the noise of the channel. In our case, we are more interested in the 'noise' that stems from the subjective semantic representations.

In the beginning of the simulation, there are no words in the vocabulary of any agent. When there is no word to express a given concept, the agents create words in a language that has the alphabet Σ consisting of nine letters: Six consonants $\Sigma_c = (b, c, d, f, g, h)$ and three vowels $\Sigma_v = a, e, i$. Each word in the language consists of two or three repetitions of the pattern CV: a consonant followed by a vowel: $L_{patt} = (CVCV, CVCVCV)$. These choices are arbitrary, and any other language production rules could be specified. The number of words in the possible vocabulary is thus fixed, but relatively large.

7.1.2 A formal model of agent's concept space

The key concept of the communication model from Publication IX is the agent's internal view of its concepts based on learning, that is, the con-

cept space, which is based on the Conceptual spaces theory introduced in Section 3.3.4. By Gärdenfors' terminology, the quality dimension D_i is represented by the feature f_i, see Chapter 3 and Gärdenfors (2000). The dimensionalities of the concept spaces can be different for each agent. The features used by agent a_1 are f_i^1, $i = 1 \dots M$ and features used by agent a_2 are f_j^2, $j = 1 \dots N$.[1] The concept spaces of the two agents are thus M-dimensional metric space $\mathcal{C}^1$ for agent a_1, and N-dimensional metric space $\mathcal{C}^2$ for agent a_2. In addition, there exist two distance measures, d_ω and d_λ, which give distances between two points inside the concept space of one agent, and between the concept spaces of the two agents, respectively.

$$d_\omega : \mathcal{C}^i \times \mathcal{C}^i \to R \quad \text{for } i = [1, 2] \tag{7.1}$$

$$d_\lambda : \mathcal{C}^i \times \mathcal{C}^j \to R \quad \text{where } i \neq j \tag{7.2}$$

While it is possible to measure d_ω in a metric space, distance measures d_λ are not easily defined.

Each agent has its own vocabulary in a form of a symbol space: $\mathcal{S}^1$ and $\mathcal{S}^2$ for agent a_1 and agent a_2 respectively. The symbols are mapped to the concept space of the agent through a mapping function ξ^i which maps the symbol $s^i \in \mathcal{S}$ to $\mathcal{C}^i$.

When communicating, agent a_i expresses a symbol $s^i \in \mathcal{S}^i$ as a signal d in the signal space $\mathcal{D}$, which is multidimensional, continuous and shared between the agents. Each agent has an individual mapping function ϕ^i from its vocabulary to the signal space, i.e., $\phi^i : \mathcal{S}^i \to \mathcal{D}$ and an inverse mapping ϕ^{-i} from signal space to the symbol space. See also Figure 7.3 for a schematic overview.

7.1.3 Building a conceptual memory

In Publications VII and VIII, the Self-Organizing Map is used to implement the conceptual memory of an agent, following the suggestion of Gärdenfors (2000). Earlier, agent simulation applications have used, for example, feedforward multilayer networks for this purpose (Cangelosi and Parisi, 1998; Grim et al., 1999). Schyns (1991) uses a modular approach similar to the one presented here, in which the Self-Organizing Map is used for categorization of the input, coupled with a supervised system for naming. Vogt (2005) uses the space spanned by n axes of the continuous-

[1] Publication VIII uses i as the index for both agents. Here, j is used for the sake of clarity.

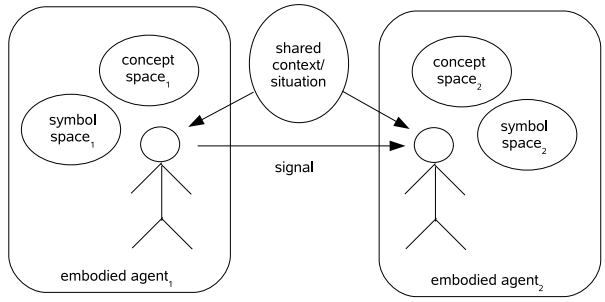
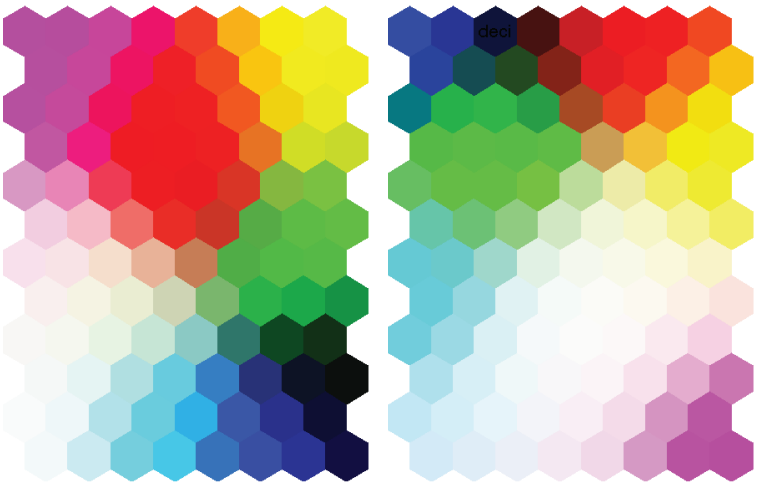


Figure 7.3. The agent communication model. From Publication IX.

valued features that are used to represent the perceived objects, in their case the color dimensions RGB and an additional shape dimension S.

The Self-Organizing Map obtains a low-dimensionality representation of the data and preserves the topological ordering (Kohonen, 2001). As a methodological alternative to the Self-Organizing Map, the Generative Topographic Map (GTM) (Bishop and Williams, 1998) could, for example, be used. Before the actual simulation process, the conceptual maps were trained separately. Figure 7.4 shows an example organization of the conceptual memories of two agents. The colors shown are the RGB values of the map vectors. The maps are well organized and transformations from one color to another are smooth. The eight prototypical colors are more



(a) Agent 1 **(b)** Agent 2

Figure 7.4. Conceptual maps of the agents from a two-agent simulation. Adapted from Publication VIII.

prominent as they are more frequent in the data than the intermediate colors that have resulted from added noise.

When an agent perceives an object, that is, the color vector, it is mapped to the Best Matching Unit (BMU) on its conceptual memory, based on the minimum distance d_ω between the perceived object vector and the map unit vectors. In the experiments of Publications VII and VIII with the SOM implementation, the Euclidean distance (cf. Table 4.3) is used as the distance measure d_ω (Eq. 7.1). If a word is observed at the same time, the word is associated to the best-matching unit the perceived vector was mapped to. Each word map-unit mapping also has an associated usage count or weight σ_i. The different schemes to define the weights are defined later in Section 7.2.3 of this chapter. A map unit can have several words associated to it and a word may be associated with several map units. In the beginning of a simulation, no map unit has any words associated with it. Through a series of language games, words become associated with the map units. Association can be many-to-many: the same word can be associated with several map units, and the same map unit can be associated with several words.

The neighborhood R_i is defined as the neighboring units of a map unit $\mathbf{m_i}$. If the size of the neighborhood $|R| = 0$, only $\mathbf{m_i}$ is considered. If $|R| = 1$, all the neighbors adjacent to $\mathbf{m_i}$ are included, and with $|R| = 2$ all adjacent neighbors of the map units belonging to the 1-neighborhood are included, see Fig. 7.5. The meaning of a word is thus defined as the map

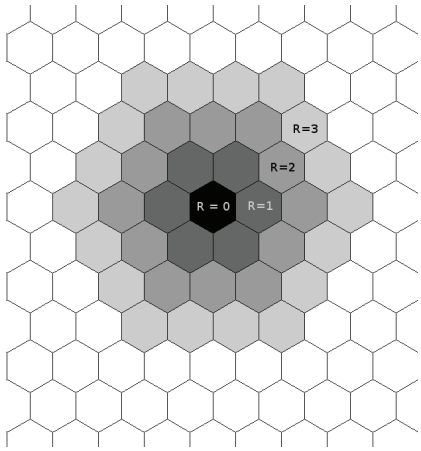


Figure 7.5. The different sizes of the neighborhood of the map unit s (black)

unit or a group of map units in the Self-Organizing Map. The word is not directly associated with the referent in the world, but to a representation, which is the best-matching unit on the map.

7.2 Modeling shared vocabulary emergence in a population

In addition to modeling individual agents, a model for communication in a population is needed.

7.2.1 The language game model revisited

To communicate, the agents engage in language games. The modeling ideas behind the language games were covered in Section 3.4.1. In Publications VII and VIII, the basic naming game algorithm based on associative learning is used:

1. Two agents are chosen randomly from a population of agents and arbitrarily given the roles of speaker and hearer.

2. The topic of the language game is randomly chosen from the set of topics.

3. Both the speaker and the hearer search for a map unit on their conceptual map that best matches the topic, i.e., find a BMU on their conceptual map.

4. The speaker searches a word that could match the topic from the R-neighborhood around the BMU, and selects a word that is best, given

a selection criterion. If no word is found, a new word is invented and associated with the BMU. Once found, this word is conveyed to the hearer.

5. The hearer then searches for a set of possible words that could denote the topic. If the word the speaker has uttered belongs to this set, the game is considered a success, else it fails. The association weights σ of both agents are updated accordingly, see Section 7.2.3 for details.

7.2.2 The two-agent communication model

A formalized version of the language game process for two agents was presented in Publication IX. As before, $\mathcal{C}^1$ and $\mathcal{C}^2$ are the conceptual space representations for agents a_1 and a_2 respectively. s^* is the symbol the Agent 1 communicates, and f^1 are the features of the context agent $a_1$observes. For a single agent,

$$s^* = \arg\min_{s \in S^1} d_\omega(f^1, \xi^1(s)) \tag{7.3}$$

The agent then selects the symbol that best corresponds to current observations by the means of some distance measure d_ω. In Publications VII and VIII, Euclidean distance is used. After symbol selection process, agent a_1 communicates the symbol $s*$ to agent a_2 expressing the symbol $s*$ in the signal space $\mathcal{D}$.

$$d = \phi^1(s^*) \tag{7.4}$$

When agent a_2 observes d, it maps it to some $s^2 \in S^2$ by using the function ϕ^{-2}. It then maps the symbol to some point in its conceptual space by using ξ^2. If this point is very near to its own observation f^2 we can say that the communication process has succeeded. That is

$$\|\xi^2(\phi^{-2}(d) - f^2)\| \leq \varepsilon, \tag{7.5}$$

where ε is a small constant and $\|\cdot\|$ is some suitable norm in $\mathcal{C}^2$.

In a two-agent model, we expect that the speaker has some estimate of the receiver's conceptual space available. This model can be learned from communication samples or it can be known *a priori*. The symbol selection process is formally given in Eq. 7.6, where $\tilde{\xi}^2$ is the model of the hearer or the receiver. The vocabulary of the hearer can also be unknown and

should be estimated by $\tilde{S}^2$.

$$s* = \arg\min_{s \in \tilde{S}^2} d_\lambda(f^1, \tilde{\xi}^2(s)) \tag{7.6}$$

This model differs considerably from the Shannon communication model, where the meaning of the messages is assumed to be known and shared.

7.2.3 Learning in language games

In the publications of this dissertation, two different algorithms were used in the learning process for the competing words for each concept.

In Publication VII, association weights σ_{w_j,m_i} are defined for each association between word w_j and map unit $\mathbf{m_i}$ for each agent k. They are updated according to the outcome of the game: increased by one when the game was successful, and decreased by one when a game failed, within upper and lower limits: $\sigma_{w_j,m_i} \in [0, 20]$. When a new association between a word and a map unit is instantiated, the weight is initialized with value $\sigma_{w_j,m_i} = 1$. The counter value approach is similar to Steels (1996), with the additional upper limit. In Vogt and Coumans (2003), the method used for naming game and guessing game is similar, but instead of decreasing the count by one the count is multiplied by a constant learning rate $\eta = 0.9$. In Vogt and Coumans (2003), lateral inhibition of competing word meaning pairs was also used. The lateral inhibition was excluded from the experiments of Publications VII and VIII, as the meanings are more fuzzy due to the neighborhood approach instead of being discrete or crisp regions.

In Publication VIII, another version of the learning function is used based on likelihood. There word-BMU association score σ_{w_j,m_i} is increased only after a successful game, and the agent selects a term to use for a given topic based on the estimated maximum likelihood. It is estimated as the number of successful uses of the term for that BMU, proportional to all of the successful uses of all the terms in that map unit.

$$p(w_j) = \frac{\sigma_{w_j,m_i}}{\sum_{j=1}^{n}(\sigma_{w_j,m_i})}. \tag{7.7}$$

The likelihood is estimated for all the terms associated with the BMU and for those map units within the neighborhood R to it. The term with the highest likelihood is selected and uttered.

7.3 Evaluation of the communication

Each of the model components can be evaluated based on the plausibility of such a component. As the purpose of the model is to find whether a shared vocabulary emerges, a number of evaluation measures are used to verify this.

Communication success

The communication or communicative success (CS) measures the outcome of the game, that is, whether the agents engaging in the language game are able to agree on the symbol use. As such, it is often incorporated as a measure in language game simulations (Vogt, 2000; De Jong, 2000). In this dissertation, it is defined as the fraction of successful games in the previous 100 games, or if fewer games are played, the fraction of successful games out of all played games.

The communication success alone is not a good measure of the quality of the emerged lexicon. It might well be that the agents just use one word to denote everything. This is why two additional measures, specificity and coherence (De Jong, 2000), have also been used.

Specificity

The specificity measure indicates the degree of the polysemy of the words in the lexicon. It decreases, if the agent uses the same word to denote different meanings. The specificity based on the preferred word is used, i.e. it is not calculated for all the words an agent could use but rather for the word that the agent would select. Specificity can be defined for each agent a_i, where $i = 1 \ldots N_a$,

$$\text{Specificity}(a_i) = \frac{(N_\rho)^2 - \sum_{k=1}^{N_\rho} f_k}{N_\rho^2 - N_\rho}, \tag{7.8}$$

where f_k is the frequency of the k^{th} word in the agent's lexicon, that is how many referents the word is associated to and N_ρ is the number of the stereotypical references. The specificity of the vocabulary of the population is defined as the mean specificity of the specificity of each individual agent.

$$\text{Specificity} = \frac{\sum_{i=1}^{n_a} \text{Specificity}(A_i)}{N_a} \tag{7.9}$$

Coherence

The coherence measure indicates the extent the agents in a population use the same word for a particular referent. Coherence thus measures the degree of synonymy in a language. For each referent $\rho_j \in R_\rho$, the stereotypical eight colors used, the words preferred by each agent are checked and the number of occurrences of the most frequent word f_{max} is divided by the number of the agents N_a (Eq. 7.10). The vocabulary coherence is a population measure obtained by averaging over all the referents (Eq. 7.11).

$$\text{Coherence}(\rho_j) = \frac{f_{max,\rho_j}}{N_a} \tag{7.10}$$

$$\text{Coherence} = \frac{\sum_{j=1}^{N_\rho} \text{Coherence}(\rho_j)}{N_\rho} \tag{7.11}$$

The specificity and coherence measures expect a static and fixed set of referents. As the current model has variation around prototypes, the prototypes have been used in calculation of these measures.

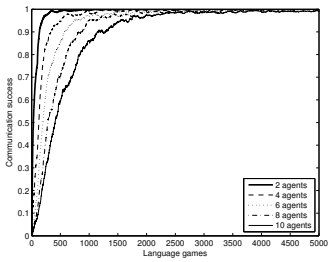
Lexicon size

The lexicon size is also measured. It defines the number of individual words in the shared lexicon. The mean lexicon size is calculated over the individual agent lexicon sizes. The lexicons of the agents also contain words that were created but never successfully used. Lexicon sizes are reported both before and after the removal of the unused words.

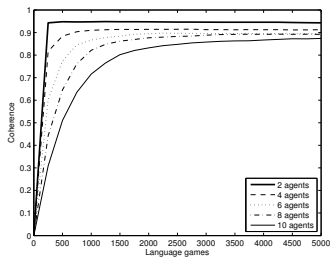
7.4 Experiments and results

This section presents the simulation results from Publications VII and VIII. The results are averaged over 10 simulation runs in both cases, experimenting with different neighborhood sizes. In Publication VII, the length of one simulation was 5 000 language games. In Publication VIII which employed the likelihood-based learning method, the convergence was slower, and simulations were run for 10 000 games. Different population sizes were tested: $N_a = [2, 4, 6, 8, 10]$ in Publication VII, and $N_a = [2, 4, 10]$ in Publication VIII.

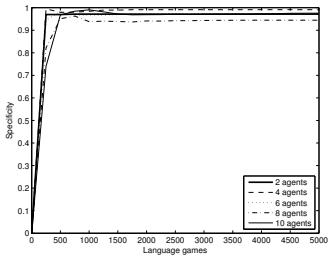
The results show that using both learning schemes for individual agent word selection, a shared lexicon emerges after a number of iterations, as indicated by the communication success scores. Figure 7.6 shows the results for an experiment with the association weight scheme from Publication VII, with a map size $M = 16 \times 12$, and neighborhood is $|R| = 2$.



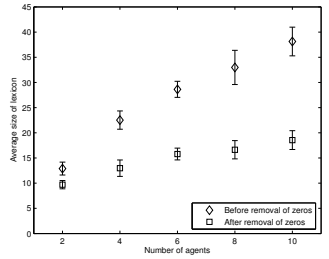
(a) Communication success



(b) Coherence



(c) Specificity



(d) Lexicon size

Figure 7.6. Simulation results for varying population sizes, with R = 2, map size $S_{map} = 16 \times 12$. From Publication VII.

The communication success score reaches the level of $CS = 0.9$ fairly fast, even when using the largest population size.

The communication success score converges slower when the population size grows. When games are played in a pairwise setting, more competing words emerge and it takes longer for the shared words to spread through a larger population, as agents have access to the same word only through subsequent games on same topic. This behavior is also illustrated in Fig. 7.6d, in which the number of unused words in the lexicon is very high for large population sizes. In the experiment of Figure 7.6, also the coherence (Fig. 7.6b) and specificity (Fig. 7.6c) scores are high, indicating a shared vocabulary with little polysemy.

In Publication VIII, a smaller map of $M = 8 \times 12$ units was used along with the maximum likelihood in the selection process. Comparing the communication success scores of the two methods, one can notice that the learning algorithm used in Publication VII converges faster and to a higher score than the likelihood-based algorithm of Publication VIII. Figure 7.8 shows the sample maps of Figure 7.4 labeled after 10 000 language games have been played. Only the most probable label for each map unit is shown. It can be seen that there are only one or two words for most prototypical colors that are preferred: 'deci' for black or dark colors, 'hihi' for blue, 'fehe' for green, 'hebe' for cyan, 'defebe' and 'gahefa' for red, 'cede' for

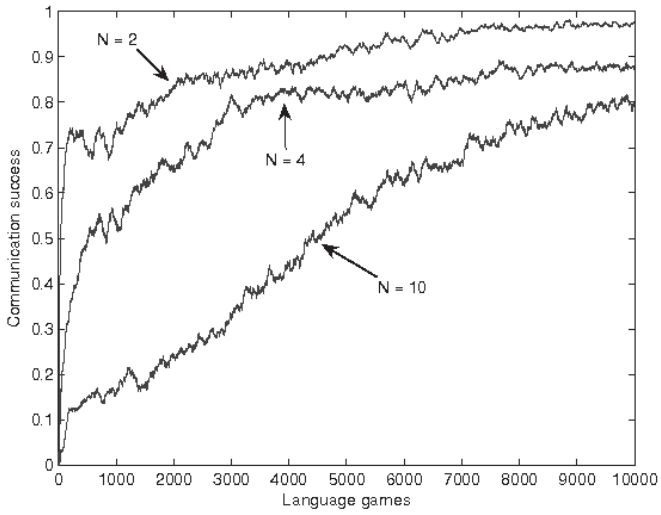


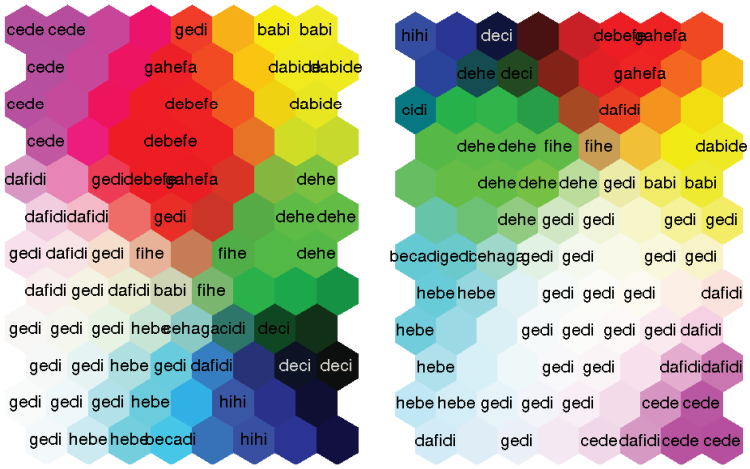
Figure 7.7. The CS scores for the likelihood-based learning algorithm. From Publication VIII.

magenta, and 'babi' and 'dabide' for yellow. For white, the most frequently used word is 'gedi', but there are also competing labels for tinted shades of white, as white covers a larger area in the space.

How does the neighborhood size affect the evaluation measures? It turns out that the CS and Coherence scores are unchanged, but there is a big difference in the Specificity scores, see Figure 7.9. With a neighborhood size $|R| = 1$, the Specificity score rises fast to $\text{Specificity} = 0.95$ after 500 games. With a neighborhood size of $|R| = 2$, the score rises fast, and then slightly drops. With $|R| = 4$, the effect is dramatic: the score only rises to $\text{Specificity} = 0.4$ and then drops to the level of $\text{Specificity} = 0.3$ as simulation advances. This indicates that only a few words are used to denote all the topics in the game.

Example maps from the same simulations are shown in Fig. 7.10, with all labels attached to the map nodes. In these visualizations, the U-matrices of the Self-Organizing Maps are shown. The shading of nodes indicates the difference between the map nodes: the darker the shade, the further apart the neighboring map units are.

In Fig. 7.10a, where $|R| = 1$, we can see that different prototypical colors are separated by the darker shades that indicate a larger distance between map nodes. All separate areas have a different label covering that area (with the exception of the area in top left, where there are several candidate labels). In Fig. 7.10b where $|R| = 4$, there is a single word,



(a) Agent 1 **(b)** Agent 2

Figure 7.8. The conceptual memories of the agents shown in Figure 7.4 after labeled with the emerged words after 10 000 language games have been played. Only the most probable label in each map node is shown. Adapted from Publication VIII.

'bihi', which covers almost all of the map. There are instances of different words present in the map, but these have not gained popularity. Thus, one can conclude that the neighborhood radius used to find candidate words depends on the map size. If the neighborhood radius is too large, single words gain popularity, and the lexicon is not specific enough. This would be alleviated if a constraint to alleviate polysemy would be in place. One such constraint could be a need to discriminate between different topics, for example, in a form of a guessing game where the agent needs to distinguish the topic of the game in a context of several objects.

The agent simulation experiments demonstrate that despite the individual subjective semantic representations, a shared vocabulary emerges in the agent population. The quality of the shared vocabulary depends largely on the size of the neighborhood size $|R|$. This approach was devised to enable a smooth transition between concepts, but one can question, whether such a fixed radius is plausible. To mimic categorization, the semantic map could be divided into separate categories by, for example, clustering the SOM in some fashion. One can also point out that while the representations are indeed grounded in (simulated) experience, the representations are simplistic, and concentrate only on naming, not considering any other functions of language. Building a simulation model which would mimic even a fraction of the richness of the human experience associated with natural language would be a very demanding task.

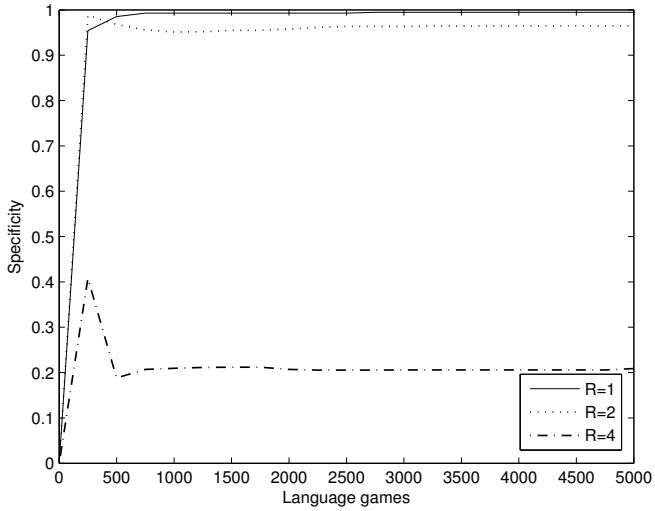
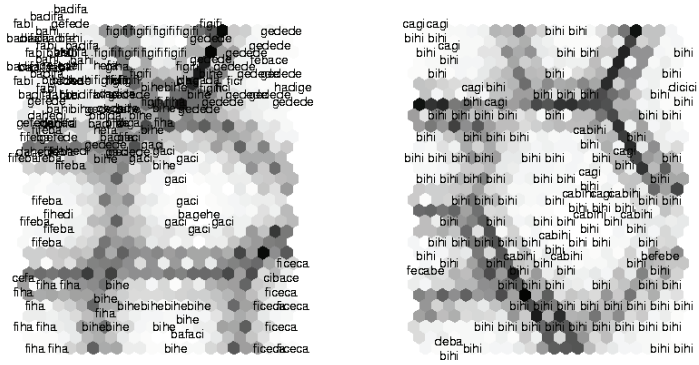


Figure 7.9. The differences in the Specificity score for different neighborhood radii in a population of six agents, using the map size $S_{map} = 16 \times 12$. The shades of gray denote distances in the original space: the larger the distance, the darker the color. Both maps have been organized to present the eight prototypical colors used in the input. From Publication VII.



(a) $|R| = 1$ **(b)** $|R| = 4$

Figure 7.10. Sample maps with different neighborhood size $|R|$. From Publication VII.

8. Summary and conclusions

This dissertation has contributed to several different topics related to computational modeling of language and meaning. It has provided an introduction of language modeling and presented several methodological choices for this purpose. It has contributed to different fields of study: analysing the similarity of languages, corpus-based lexical semantics, and modeling the emergence of language. The methodologies used were mostly based on information theory and machine learning, but the underlying conceptual discussion is largely independent of the methodological choices.

8.1 Analysis of the complexity of language

In this work, unsupervised methods were introduced to evaluate the similarity and differences of European languages at different linguistic levels, which contributes to the field of machine translation. Two different compression based approaches were used. In general, it can be concluded that the compression based approaches are a valid unsupervised tool in analyzing the differences of languages, when used carefully. In the first task, the analysis was carried out on both morphological and syntactic level separately. The resulting language ordering is coarse, but offers insight about the complexity at different levels. The compression results at the syntactic level correspond fairly well to the available linguistic analysis on syntactic flexibility and show that the compression methodology can be used for this purpose.

An error was discovered in the second compression experiment in the original publication, and new results were provided in this dissertation. The new results also show that languages in the same family are grouped close to each other. An overall similarity measure of a language may take into account very different aspects of language, but this is very dependent

of the representation: The non-Latin alphabet of Greek differs already at the level of encoding, and the languages thus differ fundamentally at that level. Thus the different alphabet hides the fact that several words and concepts in many Western languages are actually of Greek origin.

The Morfessor analysis was also carried out for several languages for the first time, and the results coincide with the compression results at the morphological level. Morfessor is based on the MDL principle of trying to find the minimal description for the language, and as such, it is also a certain kind of compression.

All of the languages analyzed in this dissertation were European languages, and as such, not a representative sample of the languages of the world. The availability of the text data in electronic form limits the analysis of those languages where no such electronic resources exists. Similarly, the lack of linguistic analyzes for evaluation is a problem.

8.2 Distributional modeling of word meaning

The distributional model for representing word meaning was introduced along with different semantic evaluation sets. This dissertation contains a considerable amount of work on vector space models, and their evaluation. In it, we demonstrated how multi-lingual semantic representations can be built, which would also benefit machine translation. In addition, vector representations for adjectives, a previously understudied syntactic category, were created.

A large contribution in this dissertation is demonstrating the usefulness of Independent Component Analysis in deriving corpus-based representations. It was shown that the independent components of the Independent Component Analysis method represent semantic information better than the latent features of the Latent Semantic Analysis. Furthermore, the performance of Independent Component Analysis and Latent Dirichlet Allocation was compared. It can be concluded that both of these methods are able to find groups of words that are semantically similar and correspond to human category judgments. Such methodologies could be used, for example, in sense induction, or representing the senses of polysemous words.

In addition to evaluating the performance of the unsupervised methods, the research also provided information about the concept of the category: some categories are easily represented with these methods, whereas oth-

ers are more difficult. In addition, it was shown that the unsupervised methods find structure beyond the provided class labels. This is an important aspect of such an analysis. We do not only want to confirm the performance of the computational methods, but also to find out where the linguistic analysis might be lacking. The use of the Self-Organizing Map as an explorative tool was demonstrated again: showing how different categories can be visualized, and relations between them analyzed through visualizations. Thus, this part of the study also gives insight on optimal approaches for generation and visualization of semantic representations.

As the method was not able to represent all the categories, it is natural to ask what kind of representations would be needed to represent the categories the method could not find easily. These categories seem to contain more polysemous or frequent words, and the vector space model was not able to distinguish between the different senses very well. In the work presented in this dissertation, a simple bag-of-words model was used, which does not take any syntactic information into account. A model where the different contexts of a word are explicitly defined such as proposed by Erk and Padó (2008) might be a suitable tool to model the different senses of the words in the 'difficult' categories. The contexts can be formed, for instance, using methods for segmenting text based on their topic distributions (Ginter et al., 2008).

Independent Component Analysis and Latent Dirichlet Allocation were both applied to the word set that contains known categories. Applying the methods on an unrestricted vocabulary yields coarser results. Thus, the data selection for unsupervised learning can be described as certain kind of supervision. Still, the experiments have shown that the methods also learn subcategories and sometimes other categorization beyond the labels of the evaluation set. Further analysis on the trade-off between using a large, unlabeled or partially labeled data, or smaller, labeled data sets could be performed. This dissertation concentrated on single lexemes, and there is more work to be done on representing multi-word constructs or more complex representations.

8.3 Modeling vocabulary emergence

This thesis discussed the modeling choices related to using simulation to build hypotheses of how language could originate. In the simulations, a shared vocabulary to describe the perceived objects in the world was

developed in a population of learners. This approach is experiential, i.e. the meaning of the words was directly related to experiences, and as such, the symbols the agents use are grounded. The model of an agent was built using a geometrical representation of concepts—utilizing the Self-Organizing Map as the model for conceptual memory.

Further, a language game model was formalized and used as a model for communication. Within this thesis, only a single language game, the naming game was used. Introducing the more complex games, such as the guessing game with several items in the context could well be implemented.

The agent simulation experiments show that despite the individual subjective semantic representations, a shared vocabulary can emerge in the population. In the current experiments, the quality of the shared vocabulary depends largely on the size of the neighborhood radius, with relation to the size of the semantic map. This kind of an approach enables a smooth transition between concepts, although a fixed search radius is an oversimplification. Introducing a game setting in which there would be a pressure to distinguish between several observations would be a natural future step.

Contrary to the distributional representations, the representations in the agent simulations are grounded in (simulated) experience. The representations in these simulations are based on simple experiences, and concentrate only on one function of language: naming. While some models of compositional language emergence exist, building a realistic simulation model with the richness of human experience associated with natural language would be very demanding. Yet, even with these limitations, the simulation model presented here highlights important aspects related to computational modeling of meaning: addressing subjective conceptual representations, learning, and symbol grounding.

Bibliography

A. Agostini and P. Avesani. On the discovery of the semantic context of queries by game-playing. In *Sixth International Conference on Flexible Query Answering Systems (FQAS-04)*, volume 3055 of *LNAI*, pages 203–216, Lyon, France, 2004. Springer-Verlag.

A. Almuhareb. *Attributes in Lexical Acquisition*. PhD thesis, University of Essex, 2006.

E. Alpaydin. *Introduction to Machine Learning*. MIT Press, 2004.

M. Andrews, G. Vigliocco, and D. Vinson. Integrating experential and distributional data to learn semantic representations. *Psychological Review*, 116(3): 463–498, 2009.

P. Avesani and A. Agostini. A peer-to-peer advertising game. In *Proceedings of the First International Conference on Service Oriented Computing (ICSOC-03)*, pages 28–42. Springer-Verlag, 2003.

B. Back, J. Toivonen, H. Vanharanta, and A. Visa. Comparing numerical data and text information from annual reports using self-organizing maps. *International Journal of Accounting Information Systems*, 2(4):249–269, 2001.

C.F. Baker, C.J. Fillmore, and J.B. Lowe. The Berkeley FrameNet project. In *COLING-ACL '98: Proceedings of the Conference.*, pages 86–90, Montreal, Canada, 1998.

D. Bakker. Flexibility and consistency in word order patterns in the languages of Europe. *Empirical Approaches to Language Typology*, 20:383–420, 1998.

M. Bane. Quantifying and measuring morphological complexity. In C.B. Chang and H.J. Haynie, editors, *Proceedings of the 26th West Coast Conference on Formal Linguistics*, pages 69–76, Somerville, MA, April 2008. Cascadilla Proceedings Project.

M. Baroni and A. Lenci. How we blessed distributional semantic evaluation. In *Proceedings of GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10, Edinburgh, UK, July 2011. Association for Computational Linguistics.

M. Baroni, S. Evert, and A. Lenci, editors. *Bridging the Gap between Semantic Theory and Computational Simulations: Proceedings of the ESSLLI Workshop on Distributional Lexical Semantics*. FOLLI, Hamburg, 2008.

M. Baroni, E. Barbu, B. Murphy, and M. Poesio. Strudel: A distributional semantic model based on properties and types. *Cognitive Science*, 34(2):222–254, 2010.

W.F. Battig and W.E. Montague. Category norms for verbal items in 56 categories: A replication and extension of the Connecticut category norms. *Journal of Experimental Psychology Monograph*, 80(3, part 2.):1–45, 1969.

D. Benedetto, E. Caglioti, and V. Loreto. Language trees and zipping. *Physical Review Letters*, 88(4), 2002.

A. Benz, G. Jäger, and R. van Rooij. An introduction to game theory for linguists. In *Game Theory and Pragmatics*. Palgrave Macmillan, Houndsmills, Basingtoke, Hampshire, 2006.

C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

C.M. Bishop and C.K.I. Williams. GTM: The generative topographic mapping. *Neural Computation*, 10:215–234, 1998.

D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(January):993–1022, 2003.

S. Brody and M. Lapata. Bayesian word sense induction. In *Proceedings of the 12th conference of the EACL*, pages 103–111, Athens, Greece, March-April 2009. Association for Computational Linguistics.

R. Brooks. Intelligence without representation. *Artificial Intelligence Journal*, 47:139–159, 1991.

A. Budanitsky and G. Hirst. Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, 2006.

J.A. Bullinaria and J.P. Levy. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39:510–526, 2007.

J.A. Bullinaria and J.P. Levy. Extracting semantic representations from word co-occurrence statistics: Stop-lists, stemming and SVD. *Behavior Research Methods*, 44, 2012.

W.L. Buntine and A. Jakulin. Discrete component analysis. In C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, editors, *SLSFS 2005, Lecture Notes in Computer Science*, volume 3640, pages 1–33. Springer, Heidelberg, 2006.

M. Burrows and D.J. Wheeler. A block-sorting lossless data compression algorithm. Technical report 124, Digital Equipment Corporation, Palo Alto, CA, May 1994.

B. Calderone. Learning phonological categories by independent component analysis. *Journal of Quantitative Linguistics*, 17(2):132–156, 2009.

A. Cangelosi and D. Parisi. The emergence of a 'language' in a population of evolving neural networks. *Connection Science*, 10(2):83–97, 1998.

A. Cangelosi and D. Parisi, editors. *Simulating the Evolution of Language*. Springer, 2002.

M. Cebrián, M. Alfonseca, and A. Ortega. Common pitfalls using the normalized compression distance: what to watch out for in a compressor. *Communications in Information and Systems*, 5(4):367–384, 2005.

A. Chagnaa, C.-Y. Ock, C.-B. Lee, and P. Jaimai. Feature extraction of concepts by independent component analysis. *International Journal of Information Processing Systems*, 3(1):33–37, 2007.

D. Chalmers. Strong and weak emergence. In P. Clayton and P. Davies, editors, *The Re-Emergence of Emergence*. Oxford University Press, 2006.

D. Chandler. *Semiotics for Beginners*. Daniel Chandler, 2000. http://dominicpetrillo.com/ed/Semiotics_for_Beginners.pdf, accessed Nov 25, 2013.

B. Chapelle, O. Schölkopf and A. Zien. *Semi-supervised learning*. MIT Press, 2006.

M.H. Christiansen and S. Kirby. Language evolution: The hardest problem in science? In Morten H. Christiansen and Simon Kirby, editors, *Language Evolution*, pages 1–15. Oxford University Press, New York, United States, 2002.

G. Chrupała. Efficient induction of probabilistic word classes with LDA. In *Proceedings of 5th International Joint Conference of Natural Language Processing*, pages 363–372, Chiang Mai, Thailand, 2011. Asian Federation of Natural Language Processing.

C. K. Chung and J. W. Pennebaker. Revealing dimensions of thinking in open-ended self-descriptions: An automated meaning extraction method for natural language. *Research in Personality*, 42(1):96–132, 2008.

K. Church and P. Hanks. Word association norms, mutual information and lexicography. In *Proceedings of the 27th Annual conference of the Association of Computational Linguistics*, pages 76–83, Vancouver, British Columbia, 1989.

R. Cilibrasi and P.M.B. Vitányi. Clustering by compression. *Information Theory, IEEE Transactions on*, 51(4):1523–1545, 2005.

J. Clark and C. Yallop. *An introduction to phonetics and phonology*. Blackwell, 1990.

S. Clark. Vector space models of lexical meaning. In S. Lappin and C. Fox, editors, *Handbook of Conteporary semantics*. Wiley-Blackwell, 2nd edition, forthcoming.

R. Collobert. Deep learning for efficient discriminative parsing. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research (JMLR)*, 12:2493–2537, 2011.

P. Comon. Independent component analysis—a new concept? *Signal Processing*, 36:287–314, 1994.

B. Comrie. *Language universals and linguistic typology: syntax and morphology*. Blackwell, Oxford, 1993.

M. Creutz and K. Lagus. Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 workshop on Morphological and phonological learning-Volume 6*, pages 21–30. Association for Computational Linguistics, 2002.

M. Creutz and K. Lagus. Unsupervised morpheme segmentation and morphology induction from text corpora. Technical Report A81, Laboratory of Computer and Information Science, Helsinki University of Technology, Espoo, 2005.

M. Creutz and K. Lagus. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing*, 4(1), January 2007.

W. Croft and D.A. Cruse. *Cognitive Linguistics*. Cambridge University Press, Cambridge, UK, 2004.

I. Dagan, L. Lee, and F. Pereira. Similarity-based models of word cooccurrence probabilities. *Machine Learning*, 34(1-3):43–69, 1999.

E. De Jong. *Autonomous Formation of Concepts and Communication*. PhD thesis, Vrije Universiteit Brussel, June 2000.

J.E. Deese. The associative structure of some common English adjectives. *Journal of Verbal Learning and Verbal Behavior*, 3(5):347–357, 1954.

G. Dinu and M. Lapata. Measuring distributional similarity in context. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 162–1172, MIT, Mass., October 2010. Association for Computational Linguistics.

L. Du, W.L. Buntine, and H. Jin. Modelling sequential text with an adaptive topic model. In *EMNLP-CoNLL*, pages 535–545. ACL, 2012.

K. Erk. Vector space models of word meaning and phrase meaning: a survey. *Language and Linguistics Compass*, 6(10):635–653, October 2012.

K. Erk and S. Padó. Structured vector space model for word meaning in context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 897–906, October 2008.

C.J. Fillmore. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences*, 280(1):20–32, 1976.

G. Finch. *How to Study Linguistics, A Guide to Understanding Language*, chapter 5. Palgrave Macmillan, 2nd edition, 2003.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: The concept revisited. *ACM Transactions on Information Systems*, 20(1):116–131, January 2002.

J.R. Firth. *Papers in Linguistics 1934-1951*. Oxford University Press, 1957.

W.N. Francis and H. Kucera. Brown corpus manual: Manual of information to accompany a standard corpus of present day edited American English. Technical report, Brown University, Providence, RI, 1964.

P. Gärdenfors. *Conceptual spaces: The Geometry of Thought*. MIT Press, 2000.

M. Gell-Mann and M. Ruhlen. The origin and evolution of word order. *Proceedings of the National Academy of Sciences of the United States of America*, 108 (42):17290–17295, 2011.

F. Ginter, H. Suominen, S. Pyysalo, and T. Salakoski. Combining hidden Markov models and latent semantic analysis for topic segmentation and labeling: Method and clinical application. In *Proceedings of SMBM 2008, the Third International Symposium on Semantic Mining in Biomedicine*, pages 37–44, 2008.

J. Goldstein. Emergence as a construct: History and issues. *Emergence*, 1(1): 49–72, 1999.

R.L. Goldstone. The role of similarity in categorization: Providing a groundwork. *Cognition*, 52:125–157, 1994.

S. Goldwater and T.L. Griffiths. A fully bayesian approach to unsupervised part-of-speech tagging. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 744–751. Association for Computational Linguistics, June 2007.

A. Gomes, R. Gudwin, C.N. El-Hani, and J. Queiroz. Towards the emergence of meaning processes in computers from Peircean semiotics. *Mind & Society*, 6 (2):173–187, 2007.

J.H. Greenberg. Research on language universals. *Annual Review of Anthropology*, 4:75–94, 1975.

G. Grefenstette. Finding the semantic similarity in raw text: The Deese antonyms. Technical Report FS-92-04, AAAI, 1992.

T. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101(Suppl 1):5228–5234, 2004.

T. L. Griffiths, J. B. Tenenbaum, and M. Steyvers. Topics in semantic representation. *Psychological Review*, 114(2):211–244, 2007.

P. Grim, T. Kokalis, A. Alai-Tafti, and N. Kilb. Evolution of communication with a spatialized genetic algorithm. *Evolution of Communication*, 3(2), 1999.

P. Grim, T. Kokalis, A. Alai-Tafti, N. Kilb, and P. St Denis. Making meaning happen. *Journal of Experimental and Theoretical Artificial Intelligence*, 16(4): 209–243, 2004.

P.D. Grünwald. A tutorial introduction to the minimum description length principle. In P.D. Grünwald, I.J. Myung, and M. Pitt, editors, *Advances in Minimum Description Length: Theory and Applications.*, chapter 1 and 2. MIT Press, April 2005.

P.D. Grünwald and P.M.B. Vitanyi. Kolmogorov complexity and information theory with an interpretation in terms of questions and answers. *Journal of Logic, Language and Information*, 12:497–529, 2003.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, March 2003.

L.K. Hansen, P. Ahrendt, and J. Larsen. Towards cognitive component analysis. In *Proceedings of AKRR'05, International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning*, pages 148–153, Espoo, Finland, June 2005.

S. Harnad. Psychophysical and cognitive aspects of categorical perception: A critical overview. In S. Harnad, editor, *Categorical Perception: The Groundwork of Cognition*. Cambridge University Press, New York, 1987.

S. Harnad. The symbol grounding problem. *Physica D*, 42:335–346, 1990.

Z. Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

S. Haykin. *Neural Networks A Comprehensive Foundation*. Simon & Schuster, New Jersey, 2nd edition, 1999.

T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

T. Honkela. Learning to understand—general aspects of using self-organizing maps in natural language processing. In *AIP Conference Proceedings*, volume 437, pages 563–576, 1998.

T. Honkela and A.M. Vepsäläinen. Interpreting imprecise expressions: Experiments with Kohonen's self-organizing maps and associative memory. In *Proceedings of the International Conference on Artificial Neural Networks, Icann91*, volume 1, pages 897–902, Helsinki, 1991.

T. Honkela, V. Pulkki, and T. Kohonen. Contextual relations of words in grimm tales analyzed by self-organizing map. In *Proceedings of International Conference on Artificial Neural Networks, ICANN-95*, pages 3–7. EC2 et Cie, 1995.

T. Honkela, A. Hyvärinen, and J.J. Väyrynen. WordICA — emergence of linguistic representations for words by independent component analysis. *Natural Language Engineering*, 16:277–308, 2010.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441 and 498–520, 1933.

D.A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of IRE*, 40(9):1098–1101, Sept 1952.

J.R. Hurford. Language mosaic and its evolution. In M. Christiansen and S. Kirby, editors, *Language Evolution*, pages 38–57. Oxford University Press, 2003.

A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997. ISSN 0899-7667. doi: http://dx.doi.org/10.1162/neco.1997.9.7.1483.

A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.

K.S. Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

P. Juola. Measuring linguistic complexity: The morphological tier. *Journal of Quantitative Linguistics*, 5(3):206–213, 1998.

P. Juola. Compression-based analysis of language complexity. In *Approaches to complexity in language workshop*, Helsinki, Finland, September 24–26 2005. Oral presentation.

P. Juola. Assessing linguistic complexity. In Matti Miestamo, Kaius Sinnemäki, and Fred Karlsson, editors, *Language Complexity: Typology, contact, change*, number 94 in Studies in Language Companion Series. John Benjamins, 2008.

P. Juola, T.M. Bailey, and E.M. Pothos. Theory-neutral system regularity measurements. In *Proceedings of the 20th Annual Conference of the Cognitive Science Society*, pages 555–560, 1998.

F. Kaplan. A new approach to class formation in multi-agent simulations of language evolution. In Y. Demazeau, editor, *Proceedings of the Third International Conference on Multi Agent Systems (ICMAS98)*, pages 158–165, Los Alamitos, CA, 1998. IEEE Computer Society.

F. Karlsson. Constraints on multiple center-embedding of clauses. *Journal of Linguistics*, 43(2):365–392, 2007.

S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. WEBSOM—self-organizing maps of document collections. *Neurocomputing*, 21(1-3):101–117, November 1998.

K. Katzner. *The languages of the world*. Routledge, 2002.

S. Kirby. Spontaneous evolution of linguistic structure: an iterated learning model of the emergence of regularity and irregularity. *IEEE Transactions on Evolutionary Computation*, 5(2):102–110, 2001.

S. Kirby. Natural language from artificial life. *Artificial Life*, 8:185–215, 2002.

P. Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *The 10th Machine Translation Summit*, pages 79–86. AAMT, 2005.

O. Kohonen, S. Virpioja, and K. Lagus. Semi-supervised learning of concatenative morphology. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 78–86, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

T. Kohonen. Self-organizing formation of topologically correct feature maps. *Biological Cybernetics*, 43(1):59–69, 1982.

T. Kohonen. *Self-Organizing maps*. Springer, Heidelberg, 3rd. edition edition, 2001.

A.N. Kolmogorov. On tables of random numbers. *Theoretical Computer Science*, 207:387–395, 1998. Reprinted from Sankhya: The Indian Journal of Statistics. Series A. Vol 25, Part 4 (1963).

K. Lagus, M. Creutz, and S. Virpioja. Latent linguistic codes for morphemes using independent component analysis. In A. Cangelosi, G. Bugmann, and R. Borisyuk, editors, *Modeling Language, Cognition and Action, Proceedings of the Ninth Neural Computation and Psychology Workshop, NCPW9*, Progress in Neural Processing, pages 139–144. World Scientific, 2005.

G. Lakoff. *Women, Fire, and Dangerous Things*. University of Chicago Press, Chicago, US, 1987.

G. Lakoff and M. Johnson. *Philosophy in the flesh: The embodied mind and its challenges to western thought*. Basic Books, New York, 1999.

T.K. Landauer and S.T. Dumais. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2):211–240, 1997.

R.W. Langacker. *Foundations of Cognitive Grammar: Theoretical Prerequisites*, volume 1. Stanford University Press, 1987.

S. Laurence and E. Margolis. Concepts and cognitive science. In E. Margolis and S. Laurence, editors, *Concepts: Core Readings*. MIT Press, Cambridge, MA, 1999.

M.P. Lewis, G.F. Simons, and C.D. Fennig, editors. *Ethnologue: Languages of the World*. SIL International, Dallas, Texas, 17th edition, 2013. Online version: http://www.ethnologue.com Accessed August 26, 2013.

M. Li and P.M.B. Vitanyi. *An introduction to Kolmogorov complexity and its applications*. Springer Verlag, 1997.

M. Li, X. Chen, X. Li, B. Ma, and P.M.B. Vitányi. The similarity metric. *Information Theory, IEEE Transactions on*, 50(12):3250–3264, 2004.

J. Locke. *An essay concerning human understanding*. Oxford University Press, 1690/1975.

W. Lowe. Towards a theory of semantic space. In J.D. Moore and K. Stenning, editors, *Proceedings of the Twenty-first Annual Meeting of the Cognitive Science Society*, pages 576–581. LEA, 2001.

C. Lyon, C.L. Nehaniv, and A. Cangelosi, editors. *Emergence of Communication and Language*. Springer, 2007.

E. Machery. *Doing without concepts*. Oxford University Press Oxford, 2009.

B. MacWhinney. Models of the emergence of language. *Annual Review of Psychology*, 49:199–227, 1998.

C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, 1999.

M.P. Marcus, B. Santorini, and M.A. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

D. Marechal and M.S.C. Thomas. Computational modeling in developmental psychology. *IEEE Transactions on Evolutionary Computation*, 11(2):137–150, 2007.

D. Marr and T. Poggio. From understanding computation to understanding neural circuitry. *Neurosciences Research Program bulletin*, 15:470–488, 1977.

P.H. Matthews. *Morphology*. Cambridge University Press, 2nd edition, 1991.

J.L. McClelland. The place of modeling in cognitive science. *Topics in Cognitive Science*, 1(1):11–38, 2009.

D.L. Medin and M.M. Schaffer. Context theory of classification learning. *Psychological Review*, 85(3):207–238, 1978.

M. Miestamo, K. Sinnemäki, and F. Karlsson, editors. *Language complexity: Typology, contact, change*. John Benjamins Publishing, 2008.

G.A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

M. Minsky. A framework for representing knowledge. In P. Winston, editor, *The Psychology of Computer Vision*. McGraw-Hill, 1975.

A. Moffat and A. Turpin. *Compression and coding algorithms*. Kluwer Academic, Boston, MA, 2002.

A.F. Morse and T. Ziemke. On the role(s) of modelling in cognitive science. *Pragmatics & Cognition*, 16(1):37–56, 2008.

G.L. Murphy. *The Big Book of Concepts*. MIT Press, Cambridge, Mass., 2004.

National Research Council (US) Committee on Frontiers at the Interface of Computing and Biology. Computational modeling and simulation as enablers for biological discovery. In Wooley J.C. and Lin H.S., editors, *Catalyzing Inquiry at the Interface of Computing and Biology*. National Academies Press, United States, 2005. Available from: http://www.ncbi.nlm.nih.gov/books/NBK25466/.

A. Newell and H.A. Simon. Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, 19(3):113–126, 1976.

D. Newman, E.V. Bonilla, and W.L. Buntine. Improving topic coherence with regularized topic models. In *Advances in Neural Information Processing Systems 24*, pages 496–504, Granada, Spain, December 2011.

Y. Niwa and Y. Nitta. Co-occurrence vectors from corpora vs. distance vectors from dictionaries. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 304–309, 1994.

K. Nybo, J. Venna, and S. Kaski. The self-organizing map as a visual neighbor retrieval method. In *Proceedings of 6th Int. Workshop on Self-Organizing Maps (WSOM '07)*, Bielefeld, Germany, 2007. Bielefeld University.

C.K. Ogden and I.A. Richards. *The meaning of meaning*. Routledge & Kegan Paul, 1972.

E. Oja. Unsupervised learning in neural computation. *Theoretical Computer Science*, 287:187–207, 2002.

S. Pado and M. Lapata. Constructing semantic space models from parsed corpora. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 126–135, Sapporo, Japan, 2003.

P. Pantel and D. Lin. Document clustering with committees. In *Proceedings of ACM Special Interest Group on Information Retrieval (SIGIR-02)*, pages 199–206, Tampere, Finland, 2002a.

P. Pantel and D. Lin. Discovering word senses from text. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining (KDD-02)*, pages 613–619, Edmonton, Canada, 2002b.

A. Papoulis. *Probability, Random Variables and Stochastic processes*. McGraw-Hill, 3rd edition, 1991. International ed.

M. Patel, J.A. Bullinaria, and J.P. Levy. Extracting semantic representations from large text corpora. In *Proceedings of Fourth Neural Computation and Psychology Workshop: Connectionist Representations*, pages 199–212. Springer, 1997.

K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(11):559–572, 1901.

C.S. Peirce. *Collected Papers*, volume 1. textlog.de, 1931. Available online at http://www.textlog.de/charles_s_peirce.html.

R. Pfeifer and C. Scheier. *Understanding Intelligence*. MIT Press, Cambridge, MA, 1999.

R. Rapp. Mining text for word senses using independent component analysis. In *Proceedings of SIAM International Conference on Data Mining 2004*, pages 422–426, 2004.

J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

H. Ritter and T. Kohonen. Self-organizing semantic maps. *Biological Cybernetics*, 61:241–254, 1989.

E. Rosch. Principles of categorization. In *Cognition and Categorization*, pages 27–48. Lawrence Erlbaum Associates, 1978.

H. Rubenstein and J.B. Goodenough. Contextual correlates of synonymy. *Computational Linguistics*, 8(10):627–633, 1965.

D.E. Rumelhart, J.L. McClelland, and the PDP Research Group. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1, Foundations. MIT Press, Cambridge, MA, 1986.

M. Sahlgren. *The Word-Space Model: using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. PhD thesis, Stocholm University, Department of Linguistics, 2006.

M. Sahlgren. The distributional hypothesis. *From Context to Meaning: distributional models of the lexicon in linguistics and cognitive science, Special issue of the Italian Journal of Linguistics/Rivista di Linguistica*, 20(1):33–53, 2008.

G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

E. Sapir. Grammarian and his language. *American Mercury*, 1:149–155, 1924.

F. de Saussure. *General Course in Linguistics*. McGraw-Hill Book Company, 1966.

H. Schütze. Word space. In *Advances in Neural Information Processing Systems 5*, pages 895–902. Morgan Kaufmann, 1993.

A. Schwering. Approaches to semantic similarity measurement for geo-spatial data: A survey. *Transactions in GIS*, 12(1):5–29, 2008.

P. Schyns. A modular neural network model of concept acquisition. *Cognitive Science*, 15:461–508, 1991.

C. Shannon. A mathematical theory of communication. *Bell System Techical journal*, 27:379–423, 623–656, 1948. Reprinted in Key Papers in the Development of Information Theory, IEEE Press, NY, 1974.

R.K. Shosted. Correlating complexity: A typological approach. *Linguistic Typology*, 10(1):1–40, 2006.

C. Spearman. The proof and measurement of association between two things. *American Journal of Psychology*, 15(1):72–101, Jan 1904.

L. Steels. Emergent adaptive lexicons. In P. Maes, M. J. Mataric, J. A. Meyer, J. Pollack, and S. W. Wilson, editors, *Proceedings of the Simulation of Adaptive Behavior Conference*, SAB96, pages 562–567, Cambridge, MA, 1996. MIT Press.

L. Steels and F. Kaplan. Situated grounded word semantics. In *Proceedings IJCAI'99 Proceedings of the 16th international joint conference on Artificial intelligence - Volume 2*, pages 862–867, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers.

L. Steels and P. Vogt. Grounding adaptive language games in robotic agents. In C. Husbands and I. Harvey, editors, *Proceedings of the Fourth European conference on Artificial Life*, Camridge, MA and London, 1997. MIT Press.

M. Steyvers and T. Griffiths. Probabilistic topic models. In Thomas Landauer, Danielle S. McNamara, Simon Dennis, and Walter Kintsch, editors, *Latent Semantic Analysis: A Road to Meaning*, pages 427–448. Laurence Erlbaum, 2007.

H. Suominen, S. Pyysalo, M. Hiissa, F. Ginter, S. Liu, D. Marghescu, T. Pahikkala, B. Back, H. Karsten, and T. Salakoski. Performance evaluation measures for text mining. In Min Song and Yi-Fang Wu, editors, *Handbook of Research on Text and Web Mining Technologies*, volume 2, pages 724–747. IGI Global, 2008.

R.S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

J.B. Tenenbaum. Bayesian modeling of human concept learning. In *Advances in Neural Information Processing systems 11*, pages 59–65. MIT Press, 1999.

J.B. Tenenbaum and T.L. Griffiths. Generalization, similarity and Bayesian inference. *Behavioral and brain sciences*, 24:629–640, 2001.

M.S.C. Thomas and J.L. McClelland. Connectionist models of cognition. In R. Sun, editor, *The Cambridge handbook of computational psychology*, pages 23–58. Cambridge University Press, Cambridge, 2008.

M. Tomasello. *The Cultural Origins of Human Communication*. Harvard University Press, 1999.

P.D. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the Twelfth European Conference on Machine Learning (EMCL2001)*, pages 491–502, Freiburg, Germany, 2001. Springer-Verlag.

P.D. Turney. Measuring semantic similarity by latent relational analysis. In *Proceedings of the 19th International Joint conference on Artificial intelligence (IJCAI-05)*, pages 1136–1141, 2005.

P.D. Turney and P. Pantel. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188, 2010.

J. Van Looveren. Multiple-word naming games. In E. Postma and M. Gyssens, editors, *Proceedings of the 11th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC 99)*, Maastricht, the Netherlands, 1999.

J. P. Van Overschelde, K. A. Rawson, and J. Dunlosky. Category norms: An update and expanded version of the Battig and Montague (1969) norms. *Journal of Memory and Language*, 50:289–335, 2004.

J.J. Väyrynen, L. Lindqvist, and T. Honkela. Sparse distributed representations for words with thresholded independent component analysis. In Jennie Si and Ron Sun, editors, *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2007)*, pages 1031–1036. IEEE, August 2007.

J. Venna and S. Kaski. Nonlinear dimensionality reduction as information retrieval. In Marina Meila and Xiaotong Shen, editors, *Proceedings of AISTATS 2007, the 11th International Conference on Artificial Intelligence and Statistics*, JMLR Workshop and Conference Proceedings, Volume 2: AISTATS. Omnipress, 2007.

J. Venna, J. Peltonen, K. Nybo, H. Aidos, and S. Kaski. Information retrieval perspective to nonlinear dimensionality reduction for data visualization. *Journal of Machine Learning Research*, 11:451–490, 2010.

S. Virpioja. *Learning Constructions of Natural Language: Statistical Models and Evaluations*. Aalto university doctoral dissertations 158/2012, Aalto University School of Science, December 2012.

S. Virpioja, M.-S. Paukkeri, A. Tripathi, T. Lindh-Knuutila, and K. Lagus. Evaluating vector space models with canonical correlation analysis. *Natural Language Engineering*, 18(3):399–436, July 2012.

P. Vogt. *Lexicon Grounding in Mobile Robots*. PhD thesis, Vrije Universiteit Brussel, November 2000.

P. Vogt. The physical symbol grounding problem. *Cognitive systems research*, 3 (3):429–457, 2002.

P. Vogt. The emergence of compositional structures in perceptually grounded language games. *Artificial Intelligence*, 167(1-2):206–242, 2005.

P. Vogt. Language evolution and robotics, issues on symbol grounding and language acquisition. In Angelo Loula, Ricardo Gudwin, and Jo ao Queiroz, editors, *Artificial Cognition Systems*, pages 176–209. Idea Group, Hershey, PA, 2006.

P. Vogt. Exploring the robustness of cross-situational learning under zipfian distributions. *Cognitive Science*, 36:726–739, 2012.

P. Vogt and H. Coumans. Investigating social interaction strategies for bootstrapping lexicon development. *Journal of Artificial Societies and Social Simulation*, 6(1), 2003.

J. Washtell. Co-dispersion: A windowless approach to lexical association. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL'09)*, pages 861–869, 2009.

J. Washtell and K. Markert. A comparison of windowless and window-based computational association measures as predictors of syntagmatic human associations. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP'09)*, pages 628–637, 2009.

W. Weaver. Some recent contributions to the mathematical theory of communication. In *The Mathematical Theory of Communication*, pages 1–28. University of Chicago Press, 1949.

T.A. Welch. A technique for high-performance data compression. *Computer*, 17 (6):8–19, 1984.

A.T. Wilson and P.A. Chew. Term weighting schemes for lda. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, pages 465–473, Los Angeles, California, June 2010.

L. Wittgenstein. *Philosophical Investigations*. The Macmillan Company, 1963.

L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

X. Zhu. Semi-supervised learning literature survey. Technical report 1530, Computer Sciences, 2008. Version from July 19,2008.

G.K. Zipf. *Human behavior and the principle of least effort*. Addison-Wesley Press, Cambridge, MA, 1949.

J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE transactions on Information Theory*, 24(5):530–536, 1978.

DISSERTATIONS IN INFORMATION AND COMPUTER SCIENCE

Aalto-DD20/2013 Pajarinen, Joni
Planning under uncertainty for large-scale problems with applications to wireless networking. 2013.

Aalto-DD29/2013 Hakala, Risto
Results on Linear Models in Cryptography. 2013.

Aalto-DD44/2013 Pylkkönen, Janne
Towards Efficient and Robust Automatic Speech Recognition: Decoding Techniques and Discriminative Training. 2013.

Aalto-DD47/2013 Reyhani, Nima
Studies on Kernel Learning and Independent Component Analysis. 2013.

Aalto-DD70/2013 Ylipaavalniemi, Jarkko
Data-driven Analysis for Natural Studies in Functional Brain Imaging. 2013.

Aalto-DD61/2013 Kandemir, Melih
Learning Mental States from Biosignals. 2013.

Aalto-DD90/2013 Yu, Qi
Machine Learning for Corporate Bankruptcy Prediction. 2013.

Aalto-DD128/2013 Ajanki, Antti
Inference of relevance for proactive information retrieval. 2013.

Aalto-DD205/2013 Lijffijt, Jefrey
Computational methods for comparison and exploration of event sequences. 2013.

Aalto-DD21/2014 Cho, Kyunghyun
Foundations and Advances in Deep Learning. 2013.

Aalto-DD 49/2014

ISBN 978-952-60-5643-2
ISBN 978-952-60-5644-9 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934
ISSN 1799-4942 (pdf)

Aalto University
School of Science
Department of Information and Computer Science
www.aalto.fi

BUSINESS +
ECONOMY

ART +
DESIGN +
ARCHITECTURE

SCIENCE +
TECHNOLOGY

CROSSOVER

**DOCTORAL
DISSERTATIONS**