THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Word Representations for Emergent Communication and Natural Language Processing

MIKAEL KÅGEBÄCK

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG

Gothenburg, Sweden 2018

Word Representations for Emergent Communication and Natural Language Processing
MIKAEL KÅGEBÄCK

ISBN 978-91-7597-831-4

Doktorsavhandlingar vid Chalmers Tekniska Högskola
Ny serie nr. 4512
ISSN 0346-718X
Technical Report No. 167D
Department of Computer Science and Engineering
Division of Data Science

Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Sweden
Telephone: +46 (0)31-772 1000

Cover:
Illustration of two agents playing the color game. The aim of the game is to develop a gounded color vocabulary by simulating aspects of language evolution.

Chalmers Reproservice
Gothenburg, Sweden 2018

To Sandra and Alva with Love.
♡

Abstract

The task of listing all semantic properties of a single word might seem manageable at first but as you unravel all the context dependent subtle variations in meaning that a word can encompass, you soon realize that precise mathematical definition of a word's semantics is extremely difficult. In analogy, humans have no problem identifying their favorite pet in an image but the task of precisely defining how, is still beyond our capabilities. A solution that has proved effective in the visual domain is to solve the problem by learning abstract representations using *machine learning*. Inspired by the success of learned representations in computer vision, the line of work presented in this thesis will explore learned word representations in three different contexts.

Starting in the domain of artificial languages, three computational frameworks for *emergent communication* between collaborating agents are developed in an attempt to study word representations that exhibit grounding of concepts. The first two are designed to emulate the natural development of discrete color words using *deep reinforcement learning*, and used to simulate the emergence of color terms that partition the continuous color spectra of visual light. The properties of the emerged color communication schema is compared to human languages to ensure its validity as a cognitive model, and subsequently the frameworks are utilized to explore central questions in cognitive science about universals in language within the semantic domain of color. Moving beyond the color domain, a third framework is developed for the less controlled environment of human faces and multi-step communication. Subsequently, as for the color domain we carefully analyze the semantic properties of the words emerged between the agents but in this case focusing on the grounding.

Turning the attention to the empirical usefulness, different types of learned word representations are evaluated in the context of *automatic document summarisation*, *word sense disambiguation*, and *word sense induction* with results that show great potential for learned word representations in natural language processing by reaching state-of-the-art performance in all applications and outperforming previous methods in two out of three applications.

Finally, although learned word representations seem to improve the performance of real world systems, they do also lack in interpretability when compared to classical hand-engineered representations. Acknowledging this, an effort is made towards constructing learned representations that regain some of that interpretability by designing and evaluating disentangled representations, which could be used to represent words in a more interpretable way in the future.

Acknowledgements

I want to thank my supervisor Devdatt Dubhashi for your inspiring ideas and for always believing in mine. You have helped me grow as a researcher and develop as a person. I also want to thank my co-supervisors, Richard Johansson for all our interesting discussions and your genuine care for other people, and Shalom Lappin for welcomed input and guidance. Further, I want to thank my examiner Gerardo Schneider and research school representative Agneta Nilson, for keeping me on the narrow path.

To my friends Fredrik Johansson and Olof Mogren, with whom I shared office, wrote papers, drank beers, and shared so many laughs. Thank you for making my time as a PhD student enjoyable and best of luck in the future. I would also like to acknowledge all my amazing collaborators: Asad Sayeed, Emilio Jorge, Amanda Nilsson, Maria Larsson, Emil Gustavsson, Hans Salomonsson, and Nina Tahmasebi. A big thanks to everyone in and around our research group: Peter, Christos, Alexander, Aristide, Morteza, Chien-Chung, Vinay, Ankani, Azam, Joel, and last but not least Prasanth who widened my perspective in so many ways. Also, I want to thank the former and the current head of division Peter Dybjer and Dag Wedelin for many interesting discussion ranging from chess to sailing to making sure I am using my vacation days, and for helping me on the administrative side I want to thank Rebecca Cyren, Eva Axelsson, and Anneli Andersson. I would like to acknowledge the project *Towards a knowledge-based culturomics* supported by a framework grant from the Swedish Research Council (2012–2016; dnr 2012-5738) for funding the research presented in this thesis.

Finally, all my love and gratitude goes to Sandra and Alva for the happiness we share and the joy that you bring into my life.

List of publications

This thesis is based on the following manuscripts.

Paper I M. Kågebäck, D. Dubhashi, and A. Sayeed (2018a). "A reinforcement-learning approach to efficient communication". *In submission to PLOS One*

Paper II M. Kågebäck, D. Dubhashi, and A. Sayeed (2018b). "DeepColor: Reinforcement Learning optimizes information efficiency and well-formedness in color name partitioning". *Proceedings of the 40th Annual Meeting of the Cognitive Science Society (CogSci)*

Paper III E. Jorge, M. Kågebäck, F. D. Johansson, and E. Gustavsson (2016). "Learning to Play Guess Who? and Inventing a Grounded Language as a Consequence". *Proceedings of the NIPS workshop on deep reinforcement learning*

Paper IV M. Kågebäck, O. Mogren, N. Tahmasebi, and D. Dubhashi (2014). "Extractive summarization using continuous vector space models". *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) at EACL*, pp. 31–39

Paper V O. Mogren, M. Kågebäck, and D. Dubhashi (2015). "Extractive Summarization by Aggregating Multiple Similarities". *Proceedings of Recent Advances in Natural Language Processing*, pp. 451–457

Paper VI M. Kågebäck and H. Salomonsson (2016). "Word Sense Disambiguation using a Bidirectional LSTM". *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex)*. Association for Computational Linguistics

Paper VII M. Kågebäck, F. D. Johansson, R. Johansson, and D. Dubhashi (2015). "Neural context embeddings for automatic discovery of word senses". *Proceedings of the NAACL-HLT*, pp. 25–32

Paper VIII M. Kågebäck and O. Mogren (2017). "Disentangled activations in deep networks". *Proceedings of the NIPS workshop on Learning Disentangled Features: from Perception to Control*

The following manuscripts have been published, but are not included in this work.

Paper IX N. Tahmasebi, L. Borin, G. Capannini, D. Dubhashi, P. Exner, M. Forsberg, G. Gossen, F. D. Johansson, R. Johansson, M. Kågebäck, O. Mogren, P. Nugues, and T. Risse (2015). Visions and open challenges for a knowledge-based culturomics. *International Journal on Digital Libraries* **15**.2-4, 169–187

Paper X

M. Larsson, A. Nilsson, and M. Kågebäck (2017). "Disentangled representations for manipulation of sentiment in text". *Proceedings of the NIPS workshop on Learning Disentangled Features: from Perception to Control*

Contribution summary

Paper I	Main author, designed and implemented the RL framework, contributed towards the research questions investigated, ran all the experiments, and wrote the sections describing the framework and the results in the manuscript.
Paper II	Main author, designed and implemented the RL framework, contributed towards the research questions investigated, ran all the experiments, and wrote the sections describing the framework and the results in the manuscript.
Paper III	Initiated the project, formulated the research question, contributed towards the designed of the model, and wrote abstract, introduction, and conclusions in the manuscript.
Paper IV	Main author, developed the main technical contribution, and wrote about 80% of the manuscript and experiments.
Paper V	Second author, contributed multiplicative interaction between kernels, wrote about 20% of the manuscript, and implemented the experiments related to word embeddings.
Paper VI	Developed the main technical contribution and wrote 90% of the manuscript.
Paper VII	Main author, developed the main technical contribution, and wrote about 50% of the manuscript and experiments.
Paper VIII	Initiated the project, contributed the main idea of the paper, and wrote about 50% of the manuscript and experiments.

Contents

Part I
Extended Summary

Chapter 1

Introduction

Computational understanding of language has enjoyed incredible progress over the last decade and products and services that rely on *natural language processing* now play a central role in industry as well as in people's everyday lives, providing services such as *machine translation* that bridges language barriers and brings people all over the world closer together, *question answering* systems that can outwit Jeopardy champions, and voice controlled *digital assistants* that can book you a table at your favorite restaurant. A large part of this progress can be attributed to improvements in *word representation*, i.e. mathematical representations of word semantics.

1.1 Important properties for word representations

Speakers of a language tend to have a very personal relationship to the words that make up that language. Every word has a different feel to it, that somehow encapsulates the essence of what that word means, but how do you translate this feeling into a mathematical representation that can be used in computation? When engineers design communication protocols they tend to keep the symbols orthogonal and context independent which makes the protocols compact and unambiguous. However, when humans communicate with each other they make no such effort. Instead, several words may have related or identical meaning and most words encode different senses depending on the context in which they are being used. To further add to the complexity, these idiosyncrasies follow no strict set of rules which makes human language very difficult to comprehend in an algorithmic way. Nevertheless, if computers are to understand human language these are challenges that need to be addressed. Hence, when considering the design of word representations there are a multitude of aspects that need to be taken into account and depending on specific use cases different aspects may be more or less important. However, this thesis will focus on the three properties described below.

1.1.1 Grounding in an environment

Grounding is the connection between sensory perceptions of the environment in which the agent lives and the words that the agent use (Harnad 1990). Note, this is not merely a pointer to an object but consists of the compiled experiences of an agent in respect to a word or concept. E.g. there is a large difference between knowing the lexical definition of hunger and the much deeper understanding developed while having lived through a period of famine. Grounding is important to support active use of the language as a means of communication. In this setting the language can be viewed as a tool, or an interface to other agents, that can be utilized to solve collaborative tasks. On the flip side, if agents are left to experiment and learn to solve these problems by communicating to each other, grounding could fall out as a by-product, e.g. an agent that has learned to ask for water when it is thirsty clearly now understands the effect of drinking water and this knowledge is therefore part of the agent's grounding of the word. Without that grounding, i.e. knowing what the concept of water represents for a living organism, the agent would go thirsty even if it had an extensive list of the physical properties of water and even if it had water available that it just did not know what to do with. Hence, the grounding developed in communication with others can also be useful when solving problems without a partner. Further, grounding is not a one way bridge between a text and the physical world but can also aid in the understanding of the text by generalization of concepts via properties that are rarely discussed in writing but that are obvious when seeing an object in the environment, e.g. the fact that *cars have exactly one steering wheel*, which is easy to see but not often mentioned, might help the agent understand the phrase "take the wheel".

1.1.2 Applicability to real world problems

Word representations used to be carefully constructed by people with expert knowledge. However, though the accuracy and interpretability of the information encoded in these representations were high, the challenge of creating and maintaining complex representations that cover all aspects of all words is simply insurmountable and as a result the coverage was too low for general purpose language. Instead, word representations constructed using statistics collected over vast amounts of text have taken over and have led to large improvements in a wide range of real world natural language processing tasks. Depending on how these statistical representations are constructed they fall into one of two general categories, count based representations (described more in Section 2.4), and learned representations (described in Section 2.5). Both categories are based on word co-occurrence statistics but growing evidence indicate that learned representations may lead to better performance on the end task.

1.1.3 Interpretability by disentanglement

Interpretability can be defined in several different ways but for the purpose of this thesis it is taken to mean, the extent to which all information in the word representation can be connected to some syntactic or semantic property, expressed in such a way that individual properties can be measured and manipulated. Interpretability is important since it

makes it possible to validate natural language processing systems and to optimize their performance by manually fine tuning specific properties. Learned word representations that have been trained to adhere to this notion of interpretability, are considered to be disentangled.

1.2 Research questions

The following research questions will be investigated as part of this thesis:

1 Can a multi-agent communication framework, trained using reinforcement learning, be used for investigation and hypothesis generation about cognition.

2 Does emergent communication between reinforcement learning agents result in grounded word representations?

3 Can learned word representations, computed using machine learning, be used to further improve the state-of-the-art in natural language processing?

4 Is it possible to construct more interpretable learned representations without sacrificing performance on the end task?

1.3 Thesis outline

Fundamental concepts, that provide background to the research presented in this thesis, are introduced in Chapter 2. Chapter 3 introduces the idea of emergent communication. Paper I and Paper II focuses on research question 1 while Paper III focus on research question 2. Next, in Chapter 4 the attention is turned to research question 3, i.e. the empirical usefulness of different types of learned word representations, evaluated in the tasks of *automatic document summarisation* in Paper IV and Paper V, *word sense disambiguation* in Paper VI, and *word sense induction* in Paper VII. Chapter 5 introduces the problem of interpretability for learned representations addressed in Paper VIII in an attempt to answer research question 4. Finally, in Chapter 6 concluding remarks are made and future work suggested.

Chapter 2

Fundamental concepts

In this chapter a number of fundamental concepts are introduced that provide background to the research presented in later chapters.

2.1 Artificial neural networks

Artificial neural networks encompass a wide range of models that have in common that they all are inspired by the neural networks that are found in animals. However, the networks used in the context of this theses are related in the sense that they are all trained in a supervised manner and all share the same training algorithm, i.e. backpropagation (Rumelhart, G. E. Hinton, and Williams 1986), and have a similar high level structure.

This category of artificial neural networks can be thought of as a trainable universal function approximator whose purpose is to approximate a mapping from data points to target values, i.e.

$$t_i = f(x_i; \theta),$$

and achieves this by manipulating the model parameters (θ) to minimize the approximation error over a training set of data points and targets $\{x_i, t_i\}_{i=1}^{N}$, where N is the size of the training set. The optimization is performed by setting up an error function that can be differentiated w.r.t the parameters θ, e.g. the squared error

$$E = \sum_{i=1}^{N} (f(x_i; \theta) - t_i)^2$$

which may be suitable if t is a scalar. The model is trained, i.e. suitable parameter values are found as

$$\theta = \underset{\theta}{\operatorname{argmin}}\, E$$

using stochastic gradient descent to minimize the error, i.e. by iteratively applying the update function

$$\theta := \theta - \lambda \frac{\partial E}{\partial \theta}$$

where λ is the learning rate and the gradient $\frac{\partial E}{\partial \theta}$ is estimated over a random mini-batch of examples sampled from the training set.

2.1.1 Feed-forward neural networks

Though the machinery described above is shared between different variants of models within the category considered, what differs is the structure of the function f. In feed-forward neural networks f, consists of stacked layers of artificial neurons where each neuron is connected to the outputs of all neurons in the preceding layer, or input for the first layer. The input signals to each neuron are modulated by model parameters (θ), simulating the synapses on biological neurons, summed, and pushed through a non-linear activation function to form the neuron's output signal, e.g. a neuron in the first layer of a feed-forward neural network takes the form

$$y = g(\theta^T \mathbf{x})$$

where d is the dimensionality of the input vector $\mathbf{x}$, and g denotes the activation function. Note, it is common practice to add a fixed dimension to the input and hidden layer outputs to provide a bias term to each neuron but this has been left out here in the interest of notational brevity. The activation function can be chosen to be one of many different forms but one common form is the logistic function, i.e.

$$g(z) = \frac{1}{1 + e^{-z}}$$

where z represents the input to the activation, i.e. the weighted sum of inputs to the neuron sometimes referred to as the *induced field*. Other common choices of activation function is the hyperbolic tangent function, and the rectified linear unit, i.e. $\max(0, z)$.

The layers of neurons not connected to the input or output of the network is called *hidden layers*. The size and number of hidden layers to use in the network are considered hyperparameters that need to be set prior to training. If no hidden layer is utilized then the model simplifies to logistic regression or, if a linear output neuron is used, linear regression. See Figure 2.1.1 for a graphical representation of one possible feed-forward neural network architecture and Figure 2.1.2 for the computational graph corresponding to the same model.

2.1.2 Recurrent neural networks

Not all problems can be described as a collection of independent and identically distributed (*i.i.d.*) labeled examples, e.g. in a sentence the individual words are clearly not independently sampled but depend on the surrounding words. Recurrent neural networks relax the *i.i.d.* assumption and instead model each data point in the sequence conditioned on the preceding data points. Hence, in this setting f takes the form

$$t_i = f(x_i, h_{i-1}; \theta)$$

where h_i is the hidden layer state corresponding to the preceding data point in the sequence, with the exception of first data point where h_0 can be taken to be zero or be

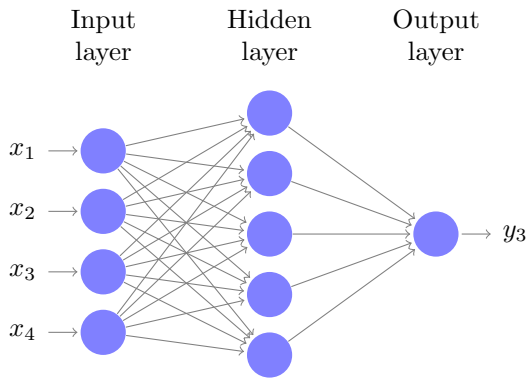


Figure 2.1.1: *Graphical representation of a feed-forward neural networkwhere each circle represents a single neuron.*

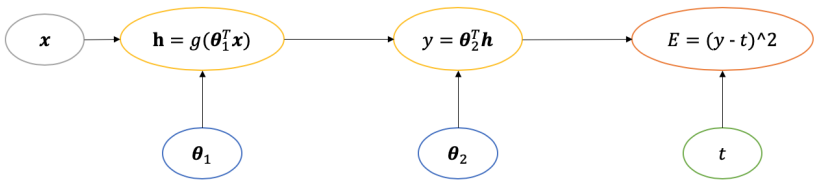


Figure 2.1.2: *Computational graph representing a feed-forward neural network with one hidden layer, a linear output layer, and a squared error cost function. Gray ovals represent input, yellow the model, blue the model parameters, red the error function, and green the labels.*

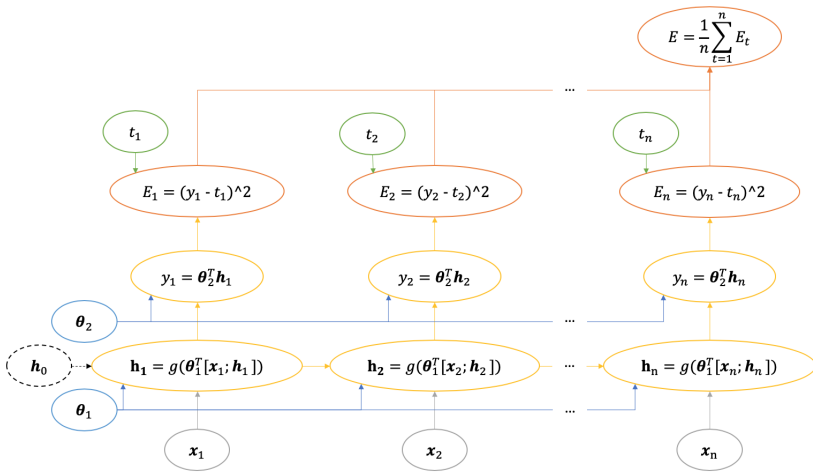


Figure 2.1.3: *Computational graph representing a recurrent neural network with one hidden layer, a linear output layer, and a squared error cost function. Gray ovals represent input, yellow the model, blue the model parameters, red the error function, and green the labels. The dashed oval can be either a fixed input or considered part of the model parameters.*

considered part of the parameters of the model and be learned during the training, see Figure 2.1.3 for the computational graph corresponding to a one hidden layer recurrent neural network.

Though the activation functions used in recurrent neural networks can be of the same type as the ones described in Section 2.1.1 it has become more common to use gated architectures that allow information to be copied forward using multiplicative gates as this helps back-propagate the gradient more efficiently over many time steps. Examples of gated architecture include the *Long short term memory* network described by Hochreiter and Schmidhuber (1997) and the *Gated recurrent networks* introduced by Cho et al. (2014).

2.2 Reinforcement learning

Reinforcement learning is a subfield to machine learning that conceptually lies between supervised learning, where labeled examples are available, and unsupervised learning which deals with raw unlabeled data and no external supervision. In reinforcement learning the model interacts with an environment that can reward or punish the decisions made by the model. However, in contrast to supervised learning the correct answer, i.e. optimal action, is never revealed to the agent and the reward may be delayed in the case of sequential problems, e.g. chess which requires a large number of actions before the reward is given in the form of a win or lose. The active part of a reinforcement learning model is referred to as the agent and the behavior of the agent is governed by the agent's

policy Π. The optimization problem that is used to train the agent attempts to find a policy that maximizes the expected cumulative future reward for the agent.

2.2.1 Deep reinforcement learning

Though the policy in general reinforcement learning can be realized in many different ways, the work in this thesis focuses on deep reinforcement learning where the policy is constructed using an artificial neural network. This type of reinforcement learning has been very successful in the context of games, mastering domains like Atari game play (Mnih, Kavukcuoglu, Silver, Rusu, et al. 2015) and the board game Go where a deep reinforcement learning agent presented by Silver, A. Huang, et al. (2016) became the first artificial intelligence to beat a world class professional Go player and subsequently, using a similar model, they were able to achieve the same without observing human play or in any other way provide human developed Go strategy. Instead the model was trained purely using self play (Silver, Schrittwieser, et al. 2017), hence, arguably emulating more than a thousand years of Go play during which the strategies used by human professional players today were developed.

A challenge with reinforcement learning, that becomes very apparent in the context of neural networks, is that the sampling of the action

$$a \sim \Pi_\theta(s)$$

where s is the state of the environment and θ the parameters of the policy, and the reward function

$$r = R(s, a)$$

are in general non-differentiable functions, which means that back-propagation can't be directly applied to maximize the reward function. Instead the model is trained by maximizing the expected reward, which can be empirically estimated over a batch of episodes. Figure 2.2.1 shows the computational graph of a policy gradient method, i.e. REINFORCE (Williams 1992), applied to a contextual bandit problem, i.e. a one step problem where the agent takes one decision based on the environment state, gets a negative or positive reward, and then the episode terminates.

Maximizing the REINFORCE objective function, J in Figure 2.2.1, can be shown to be equivalent to maximizing the expected reward. An intuitive way to see this is to consider the two parts that make up J, where the first indicates the probability that the agent would choose to do what it did again given the same state and the second the reward it received when carrying out this. Hence, the gradients of J w.r.t. the parameters θ will point in a direction that increases the probability of doing the same action again and if the reward is a large positive number, a large step will be taken in that direction during gradient ascent, however, if the reward is negative then the model will step away from this solution when following the gradient.

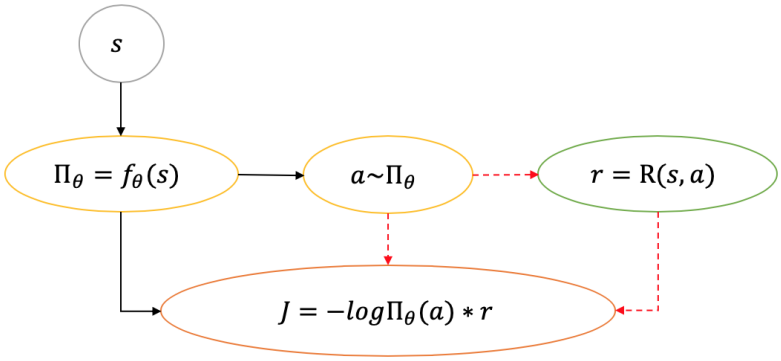


Figure 2.2.1: *Computational graph representing one episode of REINFORCE for a contextual bandit problem. A red dashed line between functions indicate a non differentiable relation. Yellow ovals represent the agent, green the environment, and red the REINFORCE objective function.*

2.3 Vector space representations

The most prevalently used structure for representing words is some kind of vector space. In fact, all representations considered in this thesis are vector space representations. Though there are several reasons to do this, the primary reason is that it provides a very general structure that can be adapted to encompass many different types of representations. Ranging all the way from hand-coded representations, where the different dimensions of the vector are used to encode different linguistic properties, to representations created using machine learning, where the model itself decides what information to encode in the vector. Another important reason for using vector space representations is to leverage all the mathematical machinery that has been developed for vector spaces, e.g. to measure the similarity between words which is often done using either euclidean distance

$$d(\mathbf{w}_a, \mathbf{w}_b) = \sqrt{\sum_{i=1}^{n}(w_{a,i} - w_{b,i})^2}$$

or cosine similarity

$$sim_{cos}(\mathbf{w}_a, \mathbf{w}_b) = \frac{\mathbf{w}_a^T \mathbf{w}_b}{\|\mathbf{w}_a\| \|\mathbf{w}_b\|}$$

between the vector representations of the words in question, here denoted a and b.

2.3.1 One-hot encodings

The most basic way of encoding a word in a vector space is called a one-hot encoding, i.e. a vector of the same dimensionality as the size of the vocabulary where all but one

dimension are zero and the remaining is one, the index of which indicates what word the vector represents. This means that the vocabulary makes up an orthonormal basis for the vector space, which has the advantage that no assumptions about the words are being encoded in their representations. This orthonormal property makes them useful in some applications, however, as semantic representations they are useless as they encode no information about the words and all words are of equal distance to each other.

2.3.2 Hand engineered word representations

To get a more semantically meaningful representation a second approach could be to list all known features of words and let them define the basis of the space. A word representation would then be a vector of zeros and ones indicating the absence or presence of corresponding word feature. However, this leads to a few problems. First, it is not clear that all features are equally important which means that you will have to weight these to make a geometric distance measure meaningful and weighting features by hand with no clear objective would be highly subjective. Second, it is simply not possible to produce the definite list of all properties of all words and even if it was possible the list would soon become outdated since language is continually changing. However, it was shown by Faruqui and Dyer (2015) that for some applications (i.e. word similarity, sentiment analysis, and NP-bracketing) state-of-the-art results can be achieved using hand engineered word representations constructed by compiling information from eight different linguistic resources, including WordNet (Miller 1995) and FrameNet (C. F. Baker, Fillmore, and Lowe 1998).

2.4 Distributional word representations

Can language statistics be used to improve the scalability and objectivity of word representations? This difficult question was answered by Harris (1954) with the *distributional hypothesis* stating that, in the words of John Rupert Firth, *You shall know a word by the company it keeps*, i.e. statistics regarding which words co-occur can be used to form *distributional word representations*. These representations are related to the one-hot encodings described in Section 2.3.1 as they can be formed by summing all one-hot vectors corresponding to tokens occurring within a given context window, e.g. three positions before and after the word in question, of a given word in a corpus. If this vector is subsequently normalized to sum to one, each dimension will indicate the probability of co-occurring with the word corresponding to that dimension, hence, words that tend to show up in similar contexts will get similar vectors. These types of representations, when trained on a sufficiently large text corpus, will cover most aspects of the word's semantics. However, a disadvantage with distributional word representations is that they are sensitive to which data they are trained on and the size of the context window that is used when computing them, where different data and settings are optimal for different applications (Sahlgren 2006). Also, they suffer from one crucial disadvantage, *the curse of dimensionality*. This is because the dimensionality of the space equals the number of words in the vocabulary, which is very high for most languages, and has been shown to render geometric distance measures ineffective for measuring similarity between words in

these models (Baroni, Dinu, and Kruszewski 2014). That said, it is possible to compute low dimensional dense representations based on distributional representations, using e.g. singular value decomposition, and it was shown by Levy, Goldberg, and Dagan (2015) that it is possible to reach state-of-the-art performance using these representations.

2.5 Learned vector representations

In an effort to overcome the dimensionality problem of distributional word representations, Bengio, Ducharme, et al. (2003) introduced a new way of leveraging co-occurrence statistics by learning to predict the context surrounding the target word using a neural network. By solving this proxy problem the network is forced into assigning similar vectors, sometimes referred to as *neural word embeddings*, for representing similar words. It is also possible to learn word representations directly for the end task, in an end-to-end fashion, given enough labeled data and computational resources but if labeled data is scarce, then word representations trained on a large corpus of unlabeled text can be used to bootstrap the system. However, since the model by Bengio, Ducharme, et al. (2003) rely on computing a distribution over all words in the vocabulary it is computationally expensive to train on a large corpus.

2.5.1 CW vectors

The first practical algorithm for deriving learned word representations was presented by Collobert and Weston (2008). This model solved the dimensionality problem by, instead of learning the probability of each word in the context of a target word, learning to differentiate between the correct target word and a random word given a context.

2.5.2 Continuous Skip-gram

However, it was with the *continuous Skip-gram* model by Mikolov, Chen, et al. (2013), released within the *Word2vec* package, that learned word representations became widely popular. The Skip-gram model is a simplified log-linear neural network that can be efficiently trained on huge amounts of data. Later the same year this model was shown by Mikolov, Yih, and Zweig (2013) to be able to capture multiple dimensions of similarity and be used to do analogy reasoning using linear vector arithmetics, e.g. $v_{king-}v_{man} + v_{woman} \approx v_{queen}$.

The model is trained to predict the context surrounding a given *target* word, see Figure 2.5.1. Each word w is represented by two vectors, one for when the word is the target, denoted $\mathbf{u}_w$, and one for when it is in the context of another word, denoted $\mathbf{v}_w$.

Following the interpretation of the negative sampling method for Skip-gram by Levy and Goldberg (2014). Let D denote the observed data, as a set of pairs of target and context words. Then, the probability of observing the pair (w_c, w_i) of a context word c and target word i in the data is,

$$p((w_c, w_i) \in D) = \frac{1}{1 + e^{-\mathbf{v}_c^T \mathbf{u}_i}} \ ,$$

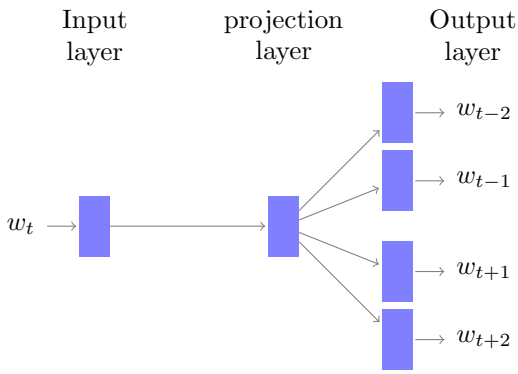


Figure 2.5.1: *The continuous Skip-gram model. Using the input word (w_t) the model tries to predict which words that will be in its context ($w_{t\pm c}$).*

where $\mathbf{u}_i$ is the vector representation of the target word w_i and $\mathbf{v}_c$ is the vector representation of the context word w_c. Training of the Skip-gram model with negative sampling corresponds to finding embeddings that maximize $p((w_c, w_i) \in D)$ for observed context pairs and $p((w_c, w_i) \notin D)$ for random (negative) context pairs. This is usually achieved using stochastic gradient descent.

2.5.3 Global vectors for word representation

Though prediction based word embeddings quickly gained interest in the community and were fast replacing the counting based distributional models, Pennington, Socher, and Manning (2014) showed that the two approaches had some complimentary properties and introduced *Global Vectors for Word Representation* (GloVe). GloVe is a hybrid approach to representing words that combine a log-linear predictive model with counting based co-occurrence statistics to more efficiently capture global statistics, something they showed was lacking in the predictive models. As such, GloVe might represent the best of both worlds.

Chapter 3

Emergent communication

In this chapter the focus is turned towards grounded word representations attained by training agents to communicate semantic concepts without any a priori shared language. Hence, to solve their task they will need to create a vocabulary of shared words, that encode useful concepts for the task, grounded in the environment that make up their world.

The idea of letting agents interact to invent their own language was pioneered by Steels (1995) but due to algorithmic and computational restraints of that time they interacted in very basic environments, which is because multi-agent communication is a very difficult task, having a state space that includes all possible conversation states and, seen from the individual agent, a non-stationary environment that continually changes while the other agents learn. However, the recent success in deep reinforcement learning (Mnih, Kavukcuoglu, Silver, Rusu, et al. 2015; Silver, A. Huang, et al. 2016), see Section 2.2.1 for details, has shown that complex tasks with huge state spaces can be learned using reinforcement learning, which has spurred renewed interest in emergent communication.

In a paper by Sukhbaatar, Szlam, and Fergus (2016) it is shown that agents can learn *continuous* communication using reinforcement learning, later used to solve tasks that require synchronisation via a global communication channel. Though promising, the continuous nature of this communication makes it very different to natural language where words are discrete symbols. This problem was addressed in a paper by J. Foerster et al. (2016) using *Differentiable inter-agent learning* (DIAL) where a method was developed to solve puzzles in a multi-agent setting where the agents were allowed to communicate by sending one-bit messages. Though their communication is still continuous during training, to enable gradient propagation, they regularize the model to promote discrete solutions and discretize during testing. This regularisation is achieved by adding noise to a continuous communication channel which has an interesting connection to the noisy-channel hypothesis (Gibson, Piantadosi, et al. 2013) providing evidence for the biological plausibility of the approach.

A first step towards grounding was taken by Lazaridou, Peysakhovich, and Baroni (2017), letting the agents play a game involving images, processed using a shared classifier trained to recognize the categories of objects present in the images. Hence, though

grounding was found it may have been a result of the supervision.

A communication model for one way communication using multiple word sentences was introduced by Havrylov and Titov (2017), which was later generalized by Mordatch and Abbeel (2017) where agents are allowed to send words back and forth freely, but this time restricting the environment to a simple grid world and sharing all parameters between agents.

3.1 Efficient communication and partitioning of color space

An important question within the study of human cognition is why different languages partition semantic spaces the way they do, e.g. though most languages have unique words for "mother" and "father", even related languages like Swedish and English have taken different stances on the parents of your parents which in Swedish will be called "mormor","morfar","farmor", or "farfar" depending on sex and side of the family while English only splits on sex. A domain that, due to its continuous nature, lends itself particularly well for the study of cognitive models for semantic partitioning is color and much research on semantic partitioning includes the analysis of the color space partitions used in different languages around the world (Regier, Kemp, and Kay 2015; Gibson, Futrell, et al. 2017; Zaslavsky, Kemp, Regier, et al. 2018; Regier, Kay, and Khetrapal 2007; Zaslavsky, Kemp, Tishby, et al. 2018; Abbott, T. L. Griffiths, and Regier 2016). Figure 3.1.1 shows an example of a color partitioning based on the Iduna language and derived based on data collected within the World Color Survey (WCS)[1], a project that compiled color naming data from 110 languages from around the world (Kay and Cook 2014). For each language, an average of 25 speakers were asked to name each color in a matrix of 330 color chips sampled from the Munsell color system to uniformly cover the human visual color spectra.

The idea that language is shaped to support *efficient communication* (Kemp, Xu, and Regier 2018; Regier, Kemp, and Kay 2015; Gibson, Futrell, et al. 2017; Piantadosi, Tily, and Gibson 2011) is an increasingly influential proposal. The theory states that language is the result of a near optimal trade-off between informativeness and complexity, i.e. given a limited vocabulary size the words will be semantically positioned for optimal informativeness.

In a recent paper by Zaslavsky, Kemp, Regier, et al. (2018) the efficient communication criterion is expressed in terms of the *information bottleneck*(IB) trade-off between complexity and accuracy. Further, they show that by manipulating the trade-off between the complexity and accuracy they can produce partitions similar to those corresponding to the languages in the WCS. Another interesting contribution in (Zaslavsky, Kemp, Regier, et al. 2018) is that they break away from the commonly used uniform prior, to describe the usefulness of different colors, and introduce two informed priors, one based on the colors of salient objects in images, and a second that is based on language statistics. Surprisingly, the image based prior performs on the same level as the uniform prior, however, the language based one significantly improves the results.

[1]The WCS data is publicly available at `http://www1.icsi.berkeley.edu/wcs/data.html`

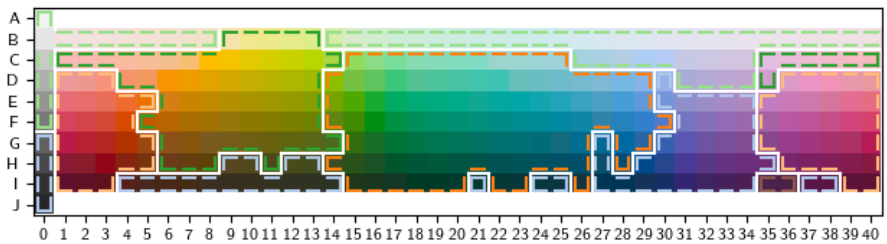


Figure 3.1.1: *WCS color map of the Iduna language, which has five color terms.*

Though the above described research provides evidence for the universality of efficient communication within human languages at their current state of development, it is, due to the nature of the real world, difficult to investigate how human languages developed over time or how the languages would have developed if the environment was different. To answer these types of questions, Paper II and Paper I introduce two communication frameworks that emulate language evolution in the controllable setting of a color communication game with agents trained using *deep reinforcement learning*, see Section 2.2.1 for a background on deep reinforcement learning. The main difference between the frameworks is found in the choice of communications channel, i.e. a discrete symbolic channel vs a noisy real valued channel.

3.1.1 The color game

The color game, illustrated in Figure 3.1.2, is a game played between two agents and the objective is to transmit color information over a bandwidth limited communications channel. At the start of each turn of the game a random color tile is drawn from a set of 330 colors. Looking at the randomly selected tile the speaking agent now has to communicate the color of the tile using a word and the listening agent subsequently needs to guess which tile the speaking agent was looking at based on the word it received. The speaking and listening agent start out *tabula rasa* with no shared language and will therefore need to build a vocabulary as they play. Further, the channel is bandwidth limited so they can not simply assign one word per tile but need to use the color structure

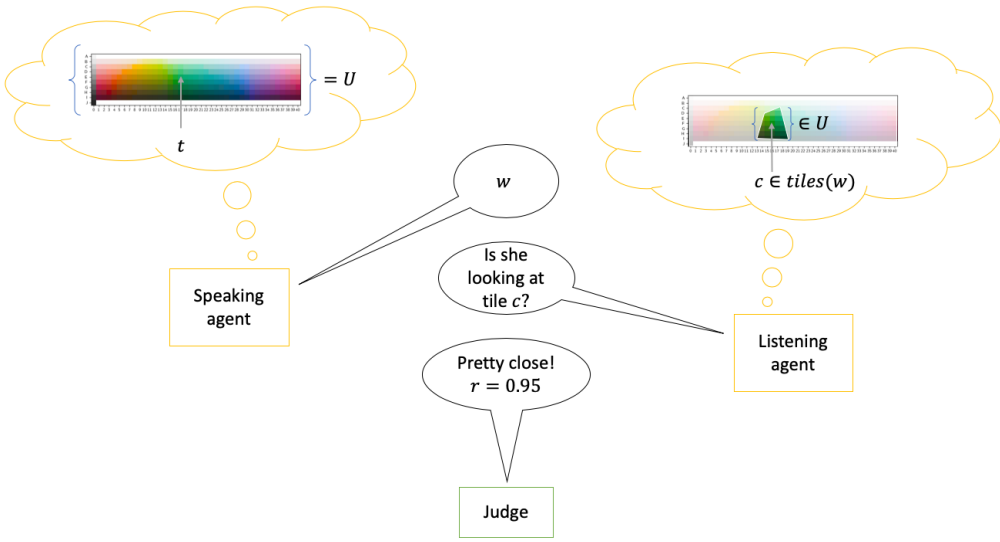


Figure 3.1.2: *Illustration the color game.*

to find an optimal vocabulary under the bandwidth restriction.

3.1.2 Frameworks

This thesis introduce two frameworks for simulating semantic partitioning through communication and validating these on the color domain by playing the color game, described in Section 3.1.1. The main difference between the two frameworks is how they model the communication channel, where the framework presented in Paper II communicates using a symbolic channel while the framework introduced in Paper I model the communication as real valued messages over a noisy channel, See Figure 3.1.3 and Figure 3.1.4 for simplified computational graphs of the two models. Further, and as a consequence of the choice of communication channel, it is interesting to note that the model in Paper II has two separate objective functions while the Paper I model use one objective for both agents. This combined objective is made possible by the real valued communication channel which is differentiable and, hence, possible to back-propagate information over, a method first introduced by J. Foerster et al. (2016). Though the biological plausibility of a differentiable communications channel is questionable, it is not unreasonable that some kind of feedback is transmitted back to the speaker from the listener to indicate if the message was understood. Further, a completely symbolic channel is also problematic with regards to biological plausibility since it is not clear that language could have ever have evolved over such a channel, i.e. isolated individuals communicating using discrete symbols would have had a hard time establishing a common grounding (in Paper II this is solved by revealing the intended object to the listener agent at the end of each turn). Nevertheless, in our experiments both these extremes seem to produce language with distinct human characteristics, which showcases the generality of the efficient communication hypothesis.

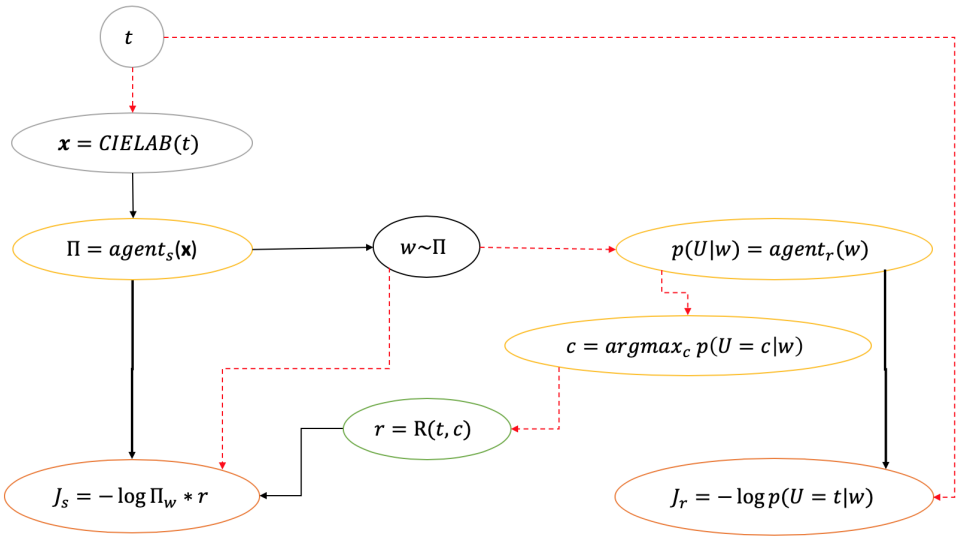


Figure 3.1.3: *Computational graph for the model in Paper II. Yellow ovals represent the agents, black the communications channel, green the reward, and red the objective function. A red dashed line between functions indicate a non differentiable relation.*

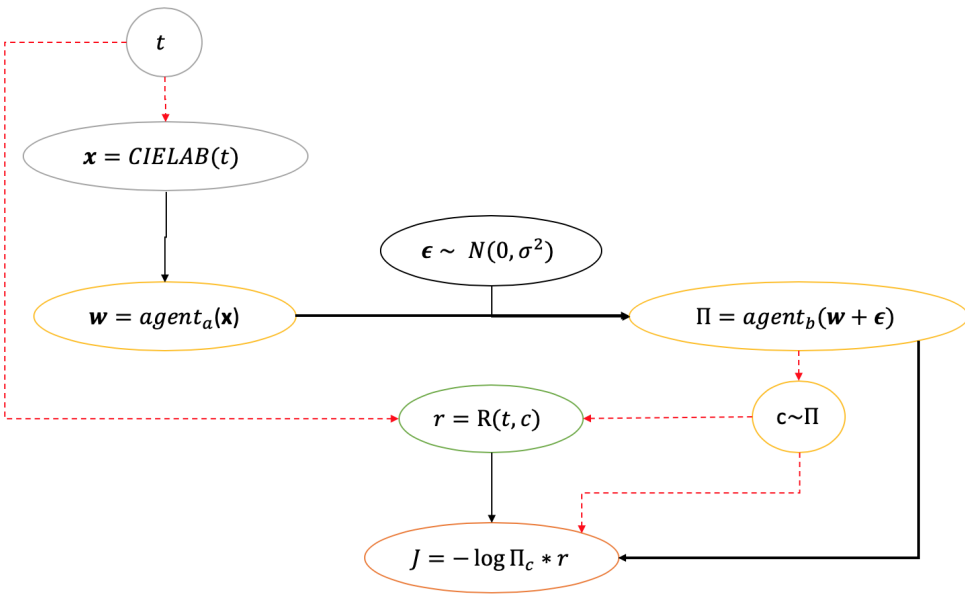


Figure 3.1.4: *Computational graph for the model in Paper I. Yellow ovals represent the agents, black the communications channel, green the reward, and red the objective function. A red dashed line between functions indicate a non differentiable relation.*

3.1.3 Model validity

The frameworks introduced above is a highly constrained version of the "real-world" scenario of many speakers negotiating meaning in a speech community. However, constrained simulations of communicative phenomena may allow the identification of plausible hypotheses about the factors that affect the corresponding real-world scenario, assuming that at least part of the expected behavior is reflected in the simulation. In Figure 3.1.5 emerged color vocabularies resulting from simulations with the framework in Paper I is compared to the human languages of the WCS in terms of efficient communication. From the figure it is clear that both human languages and the languages generated with reinforcement learning exhibit a significantly lower communication cost (i.e. KL loss) that the random baseline, neither of them reach the "optimal" communication efficiency computed using correlation clustering, and most importantly they exhibit behavior that is statistically equivalent in regards to efficient communication.

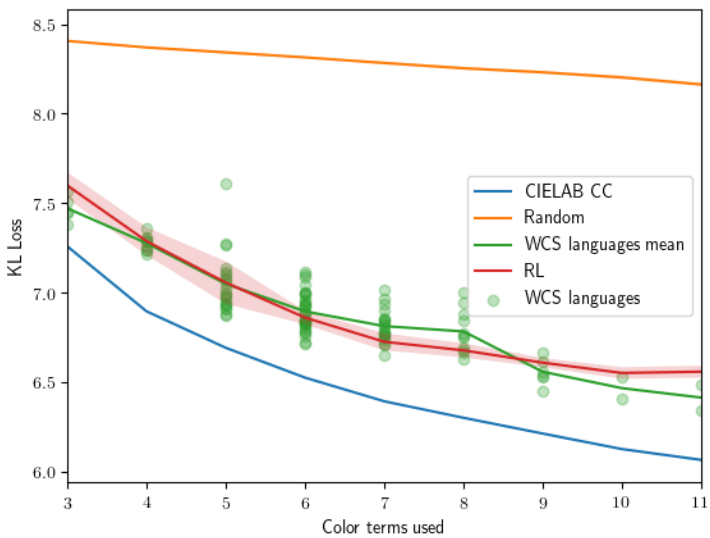


Figure 3.1.5: *KL loss at different levels of language complexity. The circles indicate the KL loss of individual WCS languages sorted based on term usage. The shaded regions indicate a 95% confidence interval*

To further investigate the similarity between generated and real languages a number of color partitionings are presented in Figure 3.1.6 depicting how human and generated languages partition the color space on average (See Paper I for a full description on how these consensus maps are constructed) using from 3 to 11 color words. It is evident that there are both similarities and differences between them. In particular, the human languages tend to have a combined green/blue area in languages with few color terms while the generated languages tend to merge light blue and pink to a greater extent, this

might be due to communicative need among humans that is not taken into account in the simulations, e.g. water, which is very important to humans, can vary between blue and green but it is seldom pink. A striking similarity between them is that they both have some relatively stable regions that seem to re-accrue while other regions subdivide.

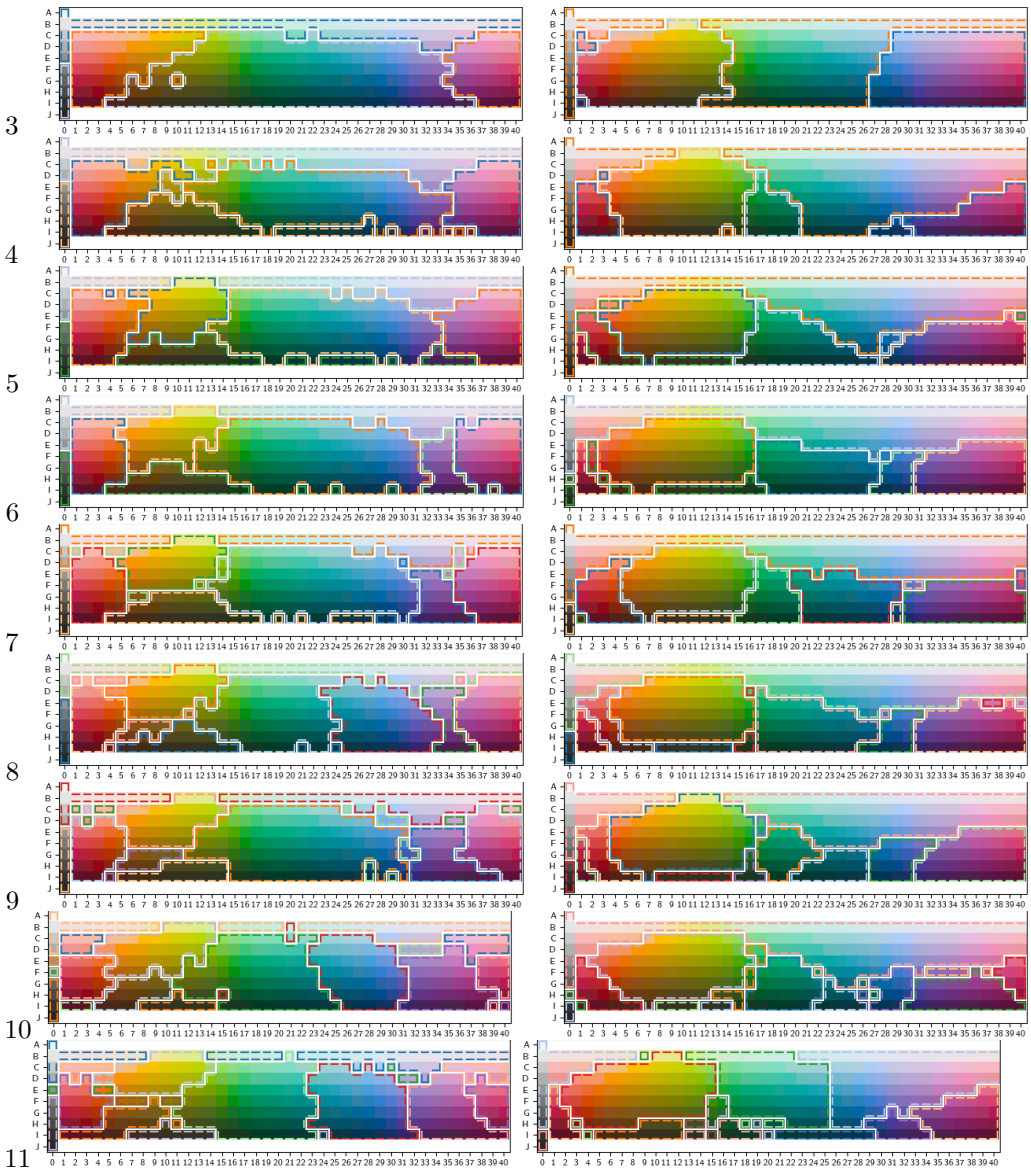


Figure 3.1.6: *Human cross-language consensus maps for 3 to 11 terms used to the left and consensus map over all reinforcement learning partitionings using 3 to 11 terms to the right.*

To complement the qualitative assessment of Figure 3.1.6, a quantitative measure of similarity, i.e. *adjusted Rand index* (Rand 1971), within and between language groups are presented in Table 3.1.1. Considering these results a corroborating image is painted showing that the within language group consistency is high for both human and generated languages while the cross group consistency is lower but far above chance indicating that they do share a similar structure.

Table 3.1.1: **Comparison of the human languages in WCS to generated languages using Rand index. Abbreviations used in table column headers: H=human, RL=reinforcement learning, and R=random**

Terms	H-H	RL-RL	H-RL	H-R
3	0.701	0.273	0.173	0.000
4	0.452	0.337	0.167	0.000
5	0.476	0.373	0.223	0.000
6	0.528	0.537	0.277	0.000
7	0.472	0.593	0.292	0.000
8	0.471	0.518	0.281	0.000
9	0.584	0.510	0.321	0.000
10	0.718	0.549	0.316	0.000
11	0.472	0.543	0.309	0.000

3.1.4 Effect of modulating the environment

A big advantage of working with a simulated environment is that it is possible to ask *"what if"* questions. An example of such a question is *"what if the color of items are not exact?"* This is an interesting question since we see that languages used by people in natural environments tend to use a fewer number of color terms, e.g. the Iduna language (see Figure 3.1.1) spoken in the Milne Bay Province of Papua New Guinea use five color terms while languages used in industrialized regions tend to utilize a larger color vocabulary, e.g. American English use ten color terms (Gibson, Futrell, et al. 2017).

In Figure 3.1.7 the resulting term usage is presented for different amount of perception noise, i.e. noise added to the color chips before showing them to the agents. Perception noise is added to simulate an environment where object of the same category may take on a range of colors, e.g. *green as a tree*, in contrast to environments with object categories of exact color, e.g. *taxi yellow*.

The results clearly indicate that perception noise does have an impact on the color vocabulary and though it is not possible to say that perception noise accounts for the full discrepancy it shows that it may be a contributing factor.

Similar results were also found in Paper II, see Figure 3.1.8 which shows that the result is robust to differences in communication channel.

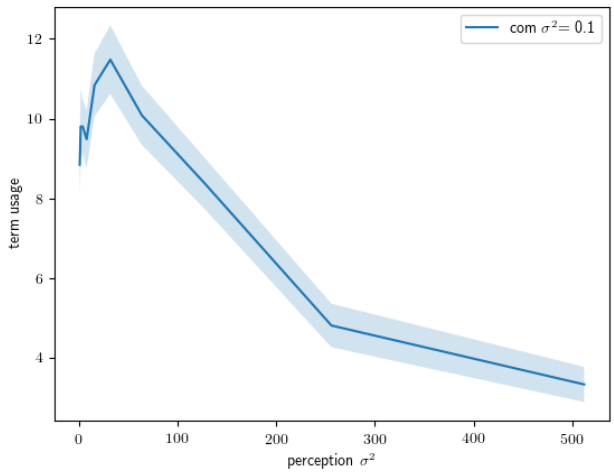


Figure 3.1.7: *The number of color terms used by the agents when different amounts of noise are applied to their perceptions.*

3.2 Complex environments and multi-step dialog

In an effort to bridge the gap between the constraint color game described in Section 3.1.1 and the real world, a more relaxed environment in the form of the children's game Guess who? is now employed. Though the game of Guess who? is still far from real life it does capture important properties that make the task more realistic (and more challenging) than the color game. Guess who? is a collaborative game, illustrated in Figure 3.2.1, where one player (the asking player) is tasked with figuring out which portrait, from a known set of portraits, that the other player (the answering player) is currently looking at. To do this the asking player gets to ask questions to which the answering player will respond yes or no, and at the end of the game the asking player will attempt a guess. When children play this game they usually play two games in parallel with opposite roles in each game and whoever first guesses correctly wins, however, for the purpose of learning grounded representations this is equivalent to the asymmetric version described above.

3.2.1 Grounding in facial properties

In contrast to Paper I and Paper II where each word correspond to non-overlapping regions of the semantic space, e.g. a specific color could be red or green but never both, Paper III deals with a higher dimension semantic space where each object corresponds to a combination of properties and each such property is encoded in the words, e.g. a portrait of a person with dark hair, glasses, and a mustache. Figure 3.2.2 shows a visualization of how the emerged words partition the space.From the figure it is clear that the model has used hair color as the primary means of partitioning. Further, though less clear, it seems

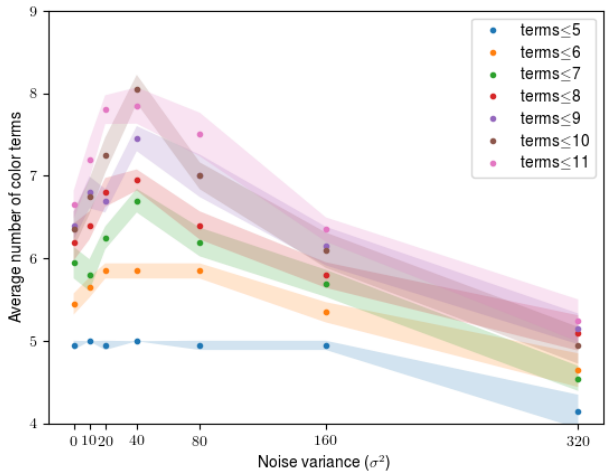


Figure 3.1.8: *Average number of words actually used by the agents after training under different amounts of noise. The points indicate the mean values from 20 independent runs and the shaded area constitutes the std. deviation $\sigma/4$.*

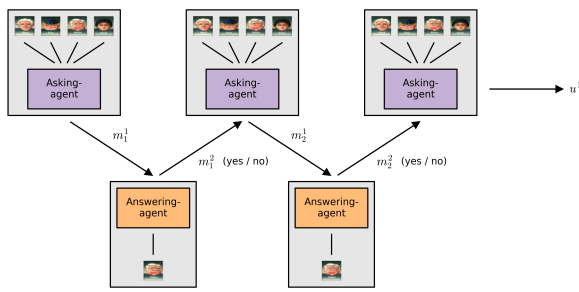


Figure 3.2.1: *Schematic illustration of our version of the Guess who? game.*

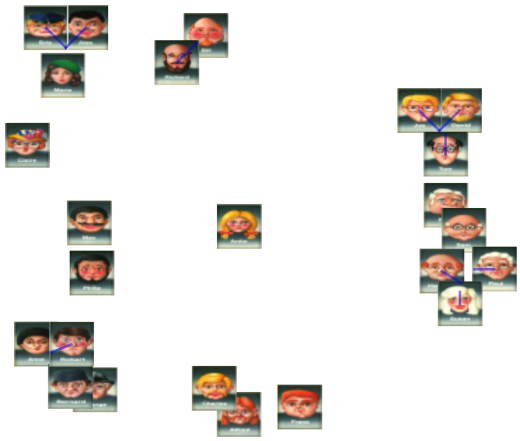


Figure 3.2.2: *The relative position of each image is determined by the bag-of-words vector describing that image. Blue lines indicate the actual position for images that have been shifted to reduce visual cluttering. The visualization was created based on the using t-SNE.*

like people with hats are concentrated to the upper left and bottom left, most faces with glasses are to the right, and that people with darker skin are mostly in the middle left area.

3.2.2 Stateful dialog

The ability to carry state over multiple steps in a back and forth dialog is notoriously difficult to replicate and a key point where many current dialog systems fail. Therefore it is noteworthy that the agents presented in Paper III have the ability to carry out multi-step conversation where the state is carried between steps and used to improve the performance of the system, see Figure 3.2.3. To the best knowledge of the author, this is the first paper to present a model that quantitatively show this ability in the context of deep reinforcement learning.

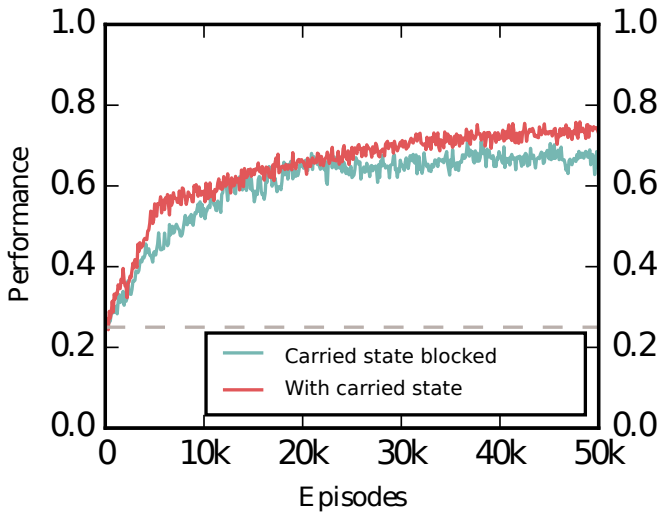


Figure 3.2.3: *Results on the Guess who? dataset with and without an incoming state for the answering agent. Eight word vocabulary, a subset of four images and two question/answers. The dashed grey line represents the baseline performance where the asking-agent guesses randomly. The results are average performance over 6 runs.*

Chapter 4

Natural language processing

In response to research question 3 defined in Section 1.2, the performance of learned word representations, described in Section 2.5, have been studied within three application areas, i.e. *automatic text summarisation*, *word sense disambiguation*, and *word sense induction*.

4.1 Automatic text summarisation

The amount of text being produced every day has exploded, which, if you want to follow what is being written on some topic is both a blessing and a curse. A blessing in that a much richer picture is being painted, less exposed to the subjective opinions of a few writers and able to cover more aspects in-depth. This sounds great, however, humans have a limited ability to read massive amounts of text, which means that you either have to limit yourself to the opinions of a handful producers or read a fair summary. However, manually producing such a summary is in most cases prohibitively expensive which is why automatic summarisation systems are becoming an increasingly important tool to keep up with the world.

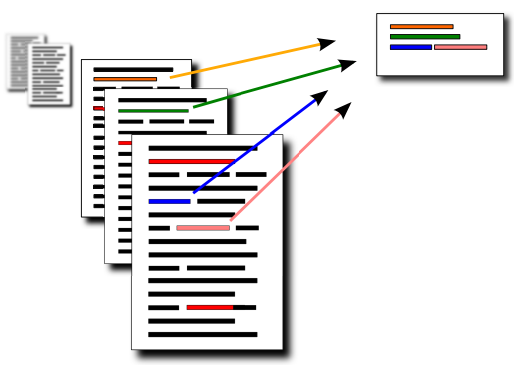


Figure 4.1.1: *Illustration of Extractive Multi-Document Summarisation.*

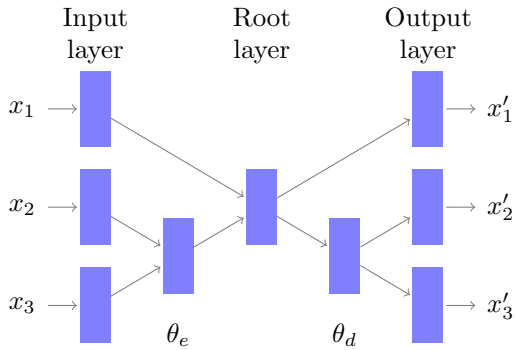


Figure 4.1.2: *The structure of an unfolding RAE, on a three word phrase ($[x_1, x_2, x_3]$). The weight matrix θ_e is used to encode the compressed representations, while θ_d is used to decode the representations and reconstruct the sentence.*

4.1.1 Extractive multi-document summarisation

Automatic summarisation of multiple documents comes in two distinct flavors, abstractive and extractive. Abstractive summarisation is the more general solution where an abstract representation of the documents is created and the summary is generated based on this representation. In contrast, extractive summarisation picks the most important sentences from the documents and put them together to form the summary, See Figure 4.1.1. Though abstractive summarisation more resembles how humans summarise text, extractive summarisation has so far been more successful at solving the task.

Using the extractive summarisation framework presented by Lin and Bilmes (2011) provides a way of extracting sentences that are both descriptive of the document set, but also diverse within the set of extracted sentences to cover as much of the information contained in the documents as possible. However, in order to perform well, this system depends on having access to a high quality sentence-to-sentence similarity measure. In Paper IV we show that learned word representations can be used to compare sentences and provide a semantically meaningful sentence-to-sentence similarity score, but to do this we have to merge the words into a sentence representation. For this we evaluate two approaches: The first is to average the representations of all words in the sentence and use this as a representation. The second approach uses a recursive auto encoder (RAE), proposed by Socher, E. H. Huang, et al. (2011) and depicted in Figure 4.1.2, to recursively merge representations guided by a parse tree and use the root layer as the sentence representation.

In Paper V we follow a similar strategy but using an expanded set of similarity measures combined with learned word representations and achieve a statistically significant improvement over the state-of-the-art on the well known dataset of *Document Understanding Conference* (DUC) 2004.

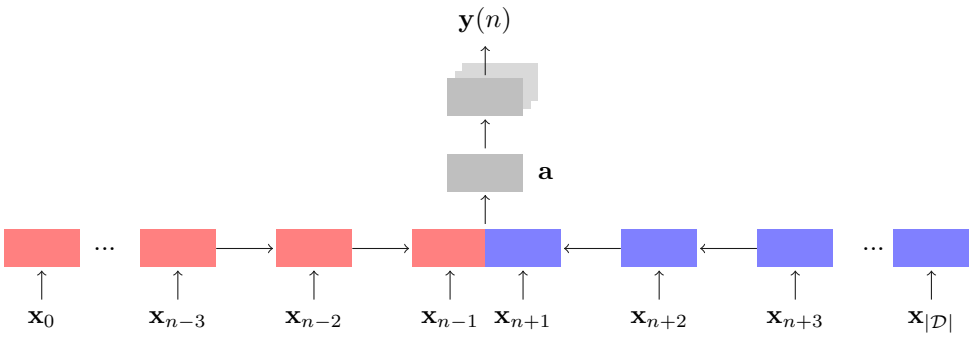


Figure 4.2.1: *A BLSTM centered around a word at position n. Its output is fed to a neural network sense classifier consisting of one hidden layer with linear units and a softmax. The softmax selects the corresponding weight matrix and bias vector for the word at position n.*

.

4.2 Word sense disambiguation

The problem of assigning a word sense, from a set of predefined senses, to a word token is referred to as *Word Sense Disambiguation* (WSD). Traditionally WSD has been approached by modeling a fixed context window surrounding the target word, i.e. the word to disambiguate, as an unordered set. Though this may work for a large set of instances it is not difficult to find examples where the order is helpful, or even necessary, for correct disambiguation. An interesting approach, overcoming this problem, was introduced by Peters et al. (2018). It uses a bidirectional language model to produce context dependent word representations and can achieve competitive results in WSD using a nearest neighbor search among prototype vectors, computed by averaging all context dependent representations belonging each sense in the training corpus.

In Paper VI an end-to-end sequence modeling approach is taken, where the order of words play an important part, and where the window is implicitly learned during training instead of defined a priori. See Figure 4.2.1 for an illustration of the model architecture.

The model stands in stark contrast to previous work in that it relies on no external features, e.g. part-of-speech taggers, parsers, knowledge graphs, etc., but still delivers results statistically equivalent to the best state-of-the-art systems. Further, we show that learned word representations play an essential role for the performance when trained on a limited amount of sense labeled data.

4.3 Word sense induction

Word Sense Induction(WSI) is the task of automatically creating a word sense inventory, i.e. lexicon, given a corpus. WSI is becoming an increasingly important tool for lexicographers trying to keep up with the ever increasing pace of language change. An approach to WSI, that was shown by Amrami and Goldberg (2018) to deliver state-of-the-art results,

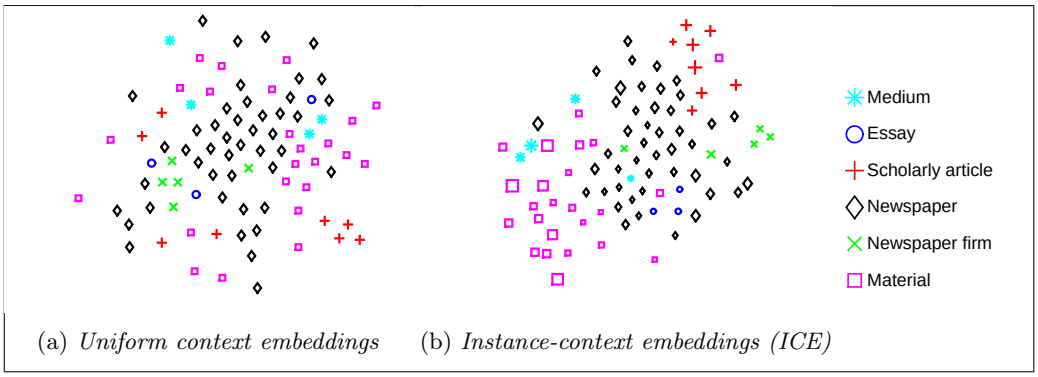
is to let a trained bi-directional language model suggest other words to substitute the target word for, at each position in the dataset. These substitute lists can subsequently be clustered, to find the different senses of the target that are word present in the dataset. To improve their results they used, what they called, dynamic symmetric patterns, i.e. they added the target word plus the word "and" to the end of the language model before predicting the substitute, and showed that this greatly improved the results.

Our approach follows the work of Schütze (1998) by employing *context clustering*, i.e. creating representations describing the context of tokens corresponding to a given word and clustering them to find the different word senses. Traditionally, the context representations used have been different variations of the distributional representations described in Section 2.4. However, in Paper VII we show that our proposed ICE representations, described below, outperforms distributional representations and achieved a relative improvement over the previous state-of-the-art method of 33% on the WSI task of SemEval-2013.

4.3.1 Instance-context embeddings

Though the learned word representations described in Section 2.5 has enjoyed much success they are actually founded on a false assumption, i.e. that each word has exactly one sense. This is clearly not true, e.g. the word *rock* may refer to either *music* or a *stone*.

In Paper VII, two approaches for computing sense aware learned word representations are introduced, where the first provide a baseline for the second approach. The baseline system constructs context dependent word representations by averaging the word representations described in Section 2.5.2 corresponding to the word tokens in their context. The drawback of the baseline approach, that we try to rectify in our second method *Instance-Context Embeddings* (ICE), is that it attends the same amount on all words in the context even though some words are clearly more indicative for deciding the sense of a given target word. Our solution to this problem is to attend more to the words to which the Skip-gram model assigns a high probability of occurring in the target words context. This means that the words that correlate with the target word will be attended to more, and that very common words that correlate with every word will be weighted less. This is due to the connection between the Skip-gram objective and *pointwise mutual information* showed in (Levy and Goldberg 2014), and has the effect of creating word representations that are more stable for word sense, see Figure 4.3.1, and less affected by the noise of unrelated words, e.g. stop words or words that are rarely used together with the target word.



(a) *Uniform context embeddings* (b) *Instance-context embeddings (ICE)*

Figure 4.3.1: *Context representations for instances of the noun "paper" in the SemEval-2013 test data, plotted using t-SNE. The legend refers to WordNet gold standard sense labels.*

Chapter 5

Towards disentangled representations

The increased performance in natural language processing, contributed to in Chapter 4 of this thesis, comes with a price in the form of decreased interpretability. Though, this might be a price that we are willing to pay in some instances there are situations in which interpretable learned word representations would be preferable, e.g. the ability to list all the properties encoded in a learned word representation would be invaluable for lexicographers. Though deep neural networks have the ability to learn representations that are increasingly abstract in deeper layers, disentangling the causes of variation in the underlying data (Bengio, Courville, and Vincent 2013), they suffer from their own power in that they are perfectly content using any arbitrarily chosen basis when encoding some property, e.g. plurality might be expressed along a line affecting all dimensions of the vector representing the word. Humans have a more limited capability when it comes to high dimensional reasoning and therefore prefer representations that represent independent factors of variation in separate dimensions of the representation, e.g. have a single dimension that encode plurality and nothing else. Another way to state this is that the representations are considered disentangled when unnecessary correlation between dimensions have been removed. Applying this notion to language, a recent paper by Subramanian et al. (2018) introduced a model for computing interpretable word representations. The model takes already trained word representations as input and use a denoising k-sparse autoencoder to learn interpretable versions them, i.e. learned word representations with disentangled dimensions.

Reflecting research question 4, defined in Section 1.2, the primary goals of the disentanglement approach presented in Paper VIII are:

- Increased interpretability of the individual dimensions in learned representations
- Compatibility with gradient based neural network models
- Computational complexity on the same level as other neural network components.

5.1 The L_Σ regularizer

The proposed solution to the goals defined above is a regularizer that actively penalizes covariance between dimensions of the representation, driving the model towards a more disentangled solution. This makes the model learn linearly uncorrelated representations which increases interpretability while obtaining good results on a number of tasks.

The regularizer is defined as

$$L_\Sigma = \frac{1}{d^2} \sum_{i,j=1}^{N} |\mathcal{C}_{ij}|$$

where d is the dimensionality of the representation layer, and

$$\mathcal{C} = \frac{1}{N-1} \sum_{i=1}^{N} (\mathbf{H} - \mathbf{1}_N \bar{\mathbf{h}})^T (\mathbf{H} - \mathbf{1}_N \bar{\mathbf{h}}).$$

is the sample covariance of the representations in the representation layer over N examples. $\mathbf{H} = [\mathbf{h}_1; ...; \mathbf{h}_N]$ is a matrix of all representations in the batch, $\mathbf{1}_N$ is an N-dimensional column vector of ones, and $\bar{\mathbf{h}}$ is the mean representation over the batch.

5.2 Selected results

Quantitative results on simulated data are presented in Figure 5.2.1 showing how the level of disentanglement influences the performance of a linear autoencoder, i.e. a neural network that is trained to recreate the input via an intermediate representation layer. The level of disentanglement is quantified using TdV which measures to what degree the total variance is captured inside the variance of the top d dimensions where d is equal to the actual dimension of the underlying data and $\text{UD}_{90\%}$ which is the number of dimensions needed to retain 90% of the total variance. The data is generated by sampling a $d = 4$ dimensional vector of independent features $z \sim N(0, \Sigma)$, where $\Sigma \in \mathcal{R}^{d \times d}$ is constrained to be non-degenerate and diagonal. However, before the data is fed to the autoencoder it is pushed through a random linear transformation $x = \Omega z$. The goal of the model is to reconstruct properties of z in the representation layer while only having access to x.

The results using L_Σ regularization are compared to results using L_1 regularization which is known to lead to sparse solutions, i.e. representations where most dimensions are exactly zero. Though L_1 does not directly penalize correlation, maximally sparse solutions can only be achieved if superfluous correlation between dimensions in the representation is removed. Hence, L_1 regularization can be seen as a crude tool for creating disentangled representations and this is seen in the result where both L_1 and L_Σ produce disentangled representations but L_Σ does so at a lower cost in performance drop.

Returning to the focus areas defined in the beginning of the section, it is clear that L_Σ regularization does improve the interpretability in the sense that fewer dimensions are used to encode the information, further evidence for this is presented in Paper VIII. Further, since the approach takes the form of an additive regularization term it can easily be added

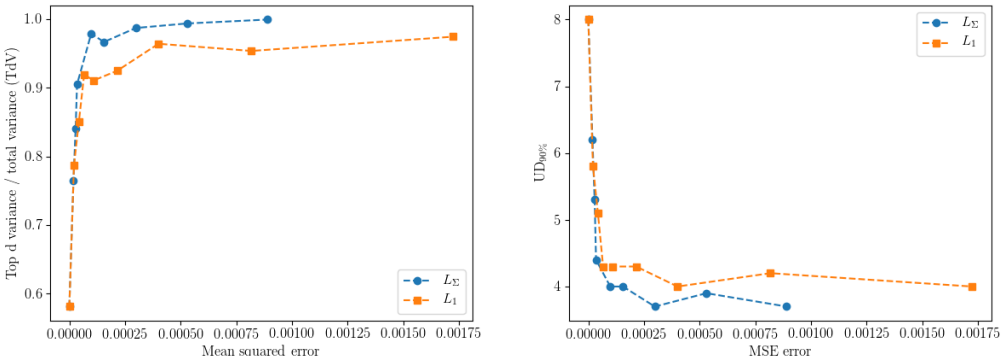


Figure 5.2.1: *The resulting dimensionality the representation layer after training the model with L_Σ and L_1 regularization respectively, measured in TdV (left) and $UD_{90\%}$ (right). The first point on each curve corresponds to $\lambda = 0$, i.e. no regularization, followed by 8 points logarithmically spaced between 0.001 and 1.All scores are averaged over 10 experiments using a different random projection (Ω).*

to any gradient based model, and finally the computational complexity is $O(Nd^2)$ which is on the same order as a single layer in a neural network ($O(Nd^{(\text{current layer})}d^{(\text{prev layer})})$) and can be efficiently computed on modern graphical processing units (GPU).

Further experiments, including disentangled representations of CIFAR-10 (Krizhevsky and G. Hinton 2009) images, is presented in Paper VIII.

Chapter 6

Concluding remarks

Following the questions posed in Section 1.2 this thesis has shown that multi-agent reinforcement learning can be used to drive language evolution in a way that produces emerged language that partition semantic spaces in human-like ways. Though close correspondence to human languages remain a goal for future work, important properties like efficient communication were shown to consistently emerge and the presented frameworks could successfully be used to model language effects of environmental manipulation. Further, the semantics of the emerged symbols were shown to be grounded in the environment, e.g. symbols connected to colors as well as more other more abstract attributes.

In the context of learned word representations, several results were presented in Chapter 4 in support of research question 3, i.e. "Can learned word representations, computed using machine learning, be used to further improve the state-of-the-art in natural language processing?", adding to the existing body of evidence showing that learned word representations can lead to clear improvements when used in natural language processing systems.

Finally, in Chapter 5 a first step was taken in the direction of learned representations with improved interpretability by showing that representations can be diagonalized by applying a covariance based regularizer, which makes it easier to connect individual dimensions to specific properties encoded in the representation, and ultimately makes the representation more interpretable.

6.1 Future work

A natural expansion to the communications frameworks presented in this theses would be to introduce multi-agent system with many participants. This would represent a more realistic setting and would make it possible to answer questions like, e.g. how term usage scales with the size of the population, the frequency and structure of inter-agent communication, and the lifespan of the individuals.

Another interesting avenue of research to follow is that of disentangled representations. In particular, an in-depth analysis of the implications of using the presented disentanglement method in the context of natural language, something that was left for future study

in the presented paper. Further, the interplay between disentangled representations and symbolic communication, which could be viewed as a form of disentanglement, would be interesting to pursue.

References

Abbott, J. T., T. L. Griffiths, and T. Regier (2016). Focal colors across languages are representative members of color categories. *Proceedings of the National Academy of Sciences* **113**.40, 11178–11183.

Amrami, A. and Y. Goldberg (2018). "Word Sense Induction with Neural biLM and Symmetric Patterns". *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 4860–4867.

Baeza-Yates, R., B. Ribeiro-Neto, et al. (1999). *Modern information retrieval*. Vol. 463. ACM press New York.

Baker, C. F., C. J. Fillmore, and J. B. Lowe (1998). "The berkeley framenet project". *Proceedings of the 17th international conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, pp. 86–90.

Baroni, M., G. Dinu, and G. Kruszewski (2014). "Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors". *Proceedings of Association for Computational Linguistics (ACL)*. Vol. 1.

Bengio, Y., A. Courville, and P. Vincent (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* **35**.8, 1798–1828.

Bengio, Y., R. Ducharme, et al. (2003). A neural probabilistic language model. *journal of machine learning research* **3**.Feb, 1137–1155.

Bordes, A. et al. (2009). "Learning to Disambiguate Natural Language Using World Knowledge". *NIPS workshop on Grammar Induction, Representation of Language and Language Learning*.

Cho, K. et al. (2014). "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation". *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pp. 1724–1734.

Collobert, R. and J. Weston (2008). "A unified architecture for natural language processing: Deep neural networks with multitask learning". *Proceedings of the 25th international conference on Machine learning*. ACM, pp. 160–167.

Faruqui, M. and C. Dyer (2015). "Non-distributional Word Vector Representations". *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Beijing, China: Association for Computational Linguistics, pp. 464–469.

Ferrucci, D. A. (2012). Introduction to "this is watson". *IBM Journal of Research and Development* **56**.3.4, 1–1.

Foerster, J. N. et al. (2016). Learning to Communicate to Solve Riddles with Deep Distributed Recurrent Q-Networks. *CoRR* **abs/1602.02672**.

Foerster, J. et al. (2016). "Learning to Communicate with Deep Multi-Agent Reinforcement Learning". *Advances in Neural Information Processing Systems 29*. Ed. by D. D. Lee et al., pp. 2137–2145.

Gibson, E., R. Futrell, et al. (2017). Color naming across languages reflects color use. *Proc Natl Acad Sci USA* **114**.40, 10785–10790.

Gibson, E., S. T. Piantadosi, et al. (2013). A noisy-channel account of crosslinguistic word-order variation. *Psychological science*.

Greenberg, J. H. (2010). *Language universals: With special reference to feature hierarchies*. Walter de Gruyter.

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena* **42**.1, 335–346.

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

Havrylov, S. and I. Titov (2017). Emergence of Language with Multi-agent Games: Learning to Communicate with Sequences of Symbols. *Workshop track at International Conference on Learning Representations (ICLR)*.

Hochreiter, S. and J. Schmidhuber (1997). Long short-term memory. *Neural computation* **9**.8, 1735–1780.

Jorge, E., M. Kågebäck, F. D. Johansson, and E. Gustavsson (2016). "Learning to Play Guess Who? and Inventing a Grounded Language as a Consequence". *Proceedings of the NIPS workshop on deep reinforcement learning*.

Kågebäck, M., D. Dubhashi, and A. Sayeed (2018a). "A reinforcement-learning approach to efficient communication". *In submission to PLOS One*.

— (2018b). "DeepColor: Reinforcement Learning optimizes information efficiency and well-formedness in color name partitioning". *Proceedings of the 40th Annual Meeting of the Cognitive Science Society (CogSci)*.

Kågebäck, M., F. D. Johansson, R. Johansson, and D. Dubhashi (2015). "Neural context embeddings for automatic discovery of word senses". *Proceedings of the NAACL-HLT*, pp. 25–32.

Kågebäck, M. and O. Mogren (2017). "Disentangled activations in deep networks". *Proceedings of the NIPS workshop on Learning Disentangled Features: from Perception to Control*.

Kågebäck, M., O. Mogren, N. Tahmasebi, and D. Dubhashi (2014). "Extractive summarization using continuous vector space models". *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) at EACL*, pp. 31–39.

Kågebäck, M. and H. Salomonsson (2016). "Word Sense Disambiguation using a Bidirectional LSTM". *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex)*. Association for Computational Linguistics.

Kay, P. and R. S. Cook (2014). World Color Survey. *Encyclopedia of Color Science and Technology*, 1–8.

Kemp, C., Y. Xu, and T. Regier (2018). Semantic Typology and Efficient Communication. *Annual Review of Linguistics* **4**.1, 109–128.

Kirby, S., T. Griffiths, and K. Smith (2014). Iterated learning and the evolution of language. *Current Opinion in Neurobiology* **28**. SI: Communication and language, 108–114.

Krizhevsky, A. and G. Hinton (2009). Learning multiple layers of features from tiny images. *Technical report, University of Toronto*.

Larsson, M., A. Nilsson, and M. Kågebäck (2017). "Disentangled representations for manipulation of sentiment in text". *Proceedings of the NIPS workshop on Learning Disentangled Features: from Perception to Control*.

Lazaridou, A., A. Peysakhovich, and M. Baroni (2017). Multi-Agent Cooperation and the Emergence of (Natural) Language. *Conference track at International Conference on Learning Representations (ICLR)*.

Levy, O. and Y. Goldberg (2014). "Neural Word Embedding as Implicit Matrix Factorization". *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., pp. 2177–2185.

Levy, O., Y. Goldberg, and I. Dagan (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics* **3**, 211–225.

Lin, H. and J. Bilmes (2011). "A Class of Submodular Functions for Document Summarization". *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. ACL, pp. 510–520.

Mikolov, T., K. Chen, et al. (2013). Efficient Estimation of Word Representations in Vector Space. *ArXiv preprint arXiv:1301.3781*.

Mikolov, T., W.-t. Yih, and G. Zweig (2013). "Linguistic regularities in continuous space word representations". *Proceedings of NAACL-HLT*, pp. 746–751.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM* **38**.11, 39–41.

Mnih, V., K. Kavukcuoglu, D. Silver, A. Graves, et al. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, et al. (2015). Human-level control through deep reinforcement learning. *Nature* **518**.7540, 529.

Mogren, O., M. Kågebäck, and D. Dubhashi (2015). "Extractive Summarization by Aggregating Multiple Similarities". *Proceedings of Recent Advances in Natural Language Processing*, pp. 451–457.

Mordatch, I. and P. Abbeel (2017). Emergence of Grounded Compositional Language in Multi-Agent Populations. *arXiv preprint arXiv:1703.04908*.

Pennington, J., R. Socher, and C. D. Manning (2014). Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)* **12**, 1532–1543.

Peters, M. E. et al. (2018). Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Piantadosi, S. T., H. Tily, and E. Gibson (2011). Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences* **108**.9, 3526–3529.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association* **66**.336, 846–850.

Regier, T., P. Kay, and N. Khetrapal (2007). Color naming reflects optimal partitions of color space. *Proc Natl Acad Sci USA* **104**.3, 1436–1441.

Regier, T., C. Kemp, and P. Kay (2015). "Word meanings across languages support efficient communication". *The handbook of language emergence*. Ed. by B. MacWhinney and W. O'Grady. Hoboken NJ: Wiley-Blackwell., pp. 237–263.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986). Learning representations by back-propagating errors. *nature* **323**.6088, 533.

Sahlgren, M. (Jan. 2006). "The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces". PhD thesis. Stockholm University.

Schütze, H. (1998). Automatic word sense discrimination. *Computational linguistics* **24**.1, 97–123.

Silver, D., A. Huang, et al. (2016). Mastering the game of Go with deep neural networks and tree search. *nature* **529**.7587, 484.

Silver, D., J. Schrittwieser, et al. (2017). Mastering the game of Go without human knowledge. *Nature* **550**.7676, 354.

Socher, R., E. H. Huang, et al. (2011). "Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection". *Advances in Neural Information Processing Systems 24*.

Socher, R., C. D. Manning, and A. Y. Ng (2010). "Learning continuous phrase representations and syntactic parsing with recursive neural networks". *Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*.

Steels, L. (1995). A self-organizing spatial vocabulary. *Artificial life* **2**.3, 319–332.

Subramanian, A. et al. (2018). SPINE: SParse Interpretable Neural Embeddings. *Proceedings of the Thirty Second AAAI Conference on Artificial Intelligence (AAAI)*.

Sukhbaatar, S., A. Szlam, and R. Fergus (2016). "Learning Multiagent Communication with Backpropagation". *Advances in Neural Information Processing Systems 29 (NIPS)*. Ed. by D. D. Lee et al., pp. 2244–2252.

Tahmasebi, N., L. Borin, G. Capannini, D. Dubhashi, P. Exner, M. Forsberg, G. Gossen, F. D. Johansson, R. Johansson, M. Kågebäck, O. Mogren, P. Nugues, and T. Risse (2015). Visions and open challenges for a knowledge-based culturomics. *International Journal on Digital Libraries* **15**.2-4, 169–187.

Williams, R. J. (1992). "Simple statistical gradient-following algorithms for connectionist reinforcement learning". *Reinforcement Learning*. Springer, pp. 5–32.

Zaslavsky, N., C. Kemp, T. Regier, et al. (2018). Efficient compression in color naming and its evolution. *Proceedings of the National Academy of Sciences*.

Zaslavsky, N., C. Kemp, N. Tishby, et al. (2018). Color naming reflects both perceptual structure and communicative need. *arXiv preprint arXiv:1805.06165*.