KU LEUVEN
FACULTEIT INGENIEURSWETENSCHAPPEN

Learning a Disease Embedding using Generalized Word2Vec Approaches.

Milan van der Meer

Thesis submitted for the degree of Master of Science in Engineering: Computer Science, specialisation Artificial Intelligence

Thesis supervisor:
Prof. dr. Roel Wuyts

Assessors:
Ir. Kn. Owsmuch
K. Nowsrest

Mentors:
Ir. An Assistent
S. Dhooghe

Academic year 2015 – 2016

Preface

I would like to thank everybody who kept me busy the last year, especially my promotor and my assistants. I would also like to thank the jury for reading the text. My sincere gratitude also goes to my wive and the rest of my family.

Milan van der Meer

Contents

Abstract

The `abstract` environment contains a more extensive overview of the work. But it should be limited to one page.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

List of Figures and Tables

List of Figures

List of Tables

List of Abbreviations and Symbols

Abbreviations

EHR	Electronic Health Record
ICD	International Classification of Diseases
WHO	World Health Organization
MedDRA	Medical Dictionary for Regulatory Activities
CBOW	Continuous Bag-of-Words

Symbols

42	"The Answer to the Ultimate Question of Life, the Universe, and Everything" according to [**?**]
c	Speed of light
E	Energy
m	Mass
π	The number pi

Chapter 1

Introduction

The first contains a general introduction to the work. The goals are defined and the modus operandi is explained.

1.1 Lorem Ipsum 4–5

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

1.2 Lorem Ipsum 6–7

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Chapter 2

Context

2.1 Introduction

In this chapter we explain the context in which we will be working for this thesis.

In section 2.2 we explain what electronic health records are and how these are represented. In section 2.3, we go further on how the electronic health records can be used to retrieve useful medical information. We explain different approaches on how to retrieve this information.

2.2 Electronic Health Records

An electronic health record (EHR) is a collection of time-stamped data about a patient over a period point of time. It is stored digitally and thus can be established for a large number of patients over a long time period.
The data stored in an EHR provides an overview of the patients health information. Health information like demographics, medical history, diagnoses, medications, and such, are stored [2].

Large countries like the US and the UK, are each investing more than 20 billion dollars into EHR systems [32]. Those systems are adopted by around 70% of the physicians. Which means a large number of physicians are using other methods or systems. Also, each country develops his own system which results in a good nationwide coverage but introduces different system around the world. We focus on disease codes in the next section and introduce two standards: one used by mainly insurance companies and the other used by pharmaceutical companies.

2.2.1 Disease Codes

To make EHRs practical it is important to adhere to standards for data formatting. A well documented standard makes it easy to store and extract information from large-scale databases of EHRs. Without the possibility of extracting information, an

EHR becomes a simple digital version of medical records on paper.
A part of an EHR consists of the diagnosis of the patient. It provides information about his disease trajectory and allows analysis on his health situation. With a uniform system for classifying diseases, it is possible to provide a general picture on health situations of populations.

ICD-10

The International Statistical Classification of Diseases and Related Health Problems (ICD) is a medical classification list made by the World Health Organization (WHO) [5]. The ICD-10 contains more than $14,400$ codes about diseases, disorders, injuries, and other related health conditions. For example, the code for a sprained ankle is $S93.4$. It also provides hierarchical categories for those codes to allow a more general overview of diseases. ICD is mainly used by insurance companies.

MedDRA

The Medical Dictionary for Regulatory Activities (MedDRA) provides medical terminology in the form of disease codes [3]. A MedDRA code is an eight digit numberic code where new terms are assigned sequentially. It does not provide clear hierarchical categories like ICD which can't be understand without a medical background. MedDRA is mainly used by pharmaceutical companies.

2.3 EHR Analytics

EHRs provide a massive amount of data which could be used to create useful insights. The data contains the medical history of a patient including medical measurements, diagnosis, prescribed drugs, and demographics. Based on those values, we could obtain the following insights:

- Effects of drugs
- Medical costs for certain diseases
- Duration and recovery percentage of certain diseases
- Correlation between demographics and certain diseases
- Link between current health state and health history
- Prediction of future health states based on history

Those insights can be offered on an individual level, which means a right intervention to the right patient at the right time. EHR analytics can be used to have a personalized care and benefits the healthcare system by cutting costs and improve outcomes.

In the following sections we talk about current EHR analytic methods.

2.3.1 Querying

Analytics in epidemiology on EHRs is typically done through querying a database [15]. A specialist can have a certain idea about correlations between conditions or patients. He can support this idea by finding cases in EHRs and analyzing the results of his query.
This method is based on the knowledge and experience of a specialist. The information has to be actively sought after and unexpected or complex correlations are not considered. Some complex relations cannot be found because of the limitations of the querying language. A query language is equivalent to first-order logic. Which means non-linear relations in the data cannot be found.

2.3.2 Big Data Analytics

More advanced methods are applied on EHR than querying. In general, they try to find patterns in the EHR data which then can be used to predict outcomes of treatments [18].

Several predictive methods from machine learning can be used and show promising results [7]. Those results are achieved by using non-optimized methods which are applied on the EHR data. We also note that methods used as Multi-layer Perceptron networks are not ideal for prediction of time-series, see section 9.4. We conclude that there is still a lot of room for improvement.

More specialized approaches are also applied on EHR data [35]. An EHR of a patient can be transformed into a matrix structure, see figure 2.1. On these matrix structures, large-scale data mining algorithms can be applied. Those make it possible to mine temporal patterns in EHR data. The found patterns can be used for prediction later on.

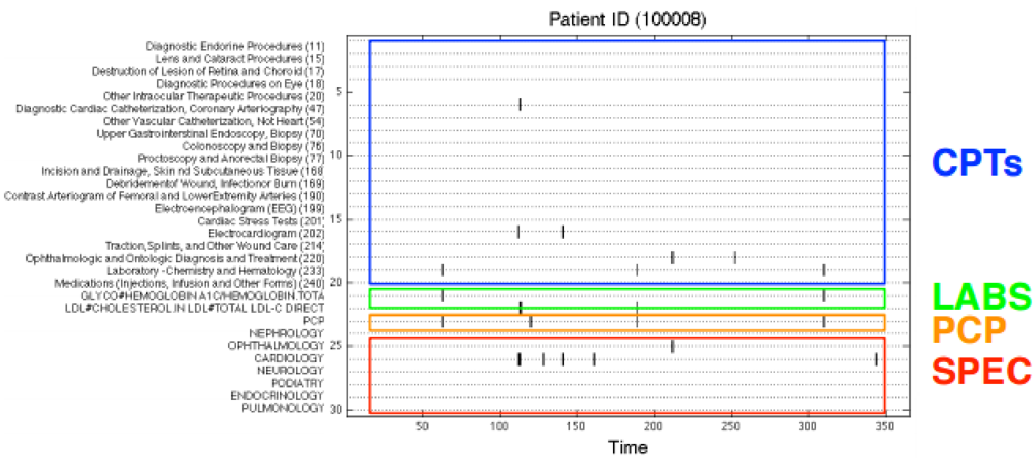


FIGURE 2.1: Example of an EHR transformed into a matrix structure [35]

It is also possible to define patient similarities [33]. So when a patient is similar to a previous patient, his treatment can be based on previous experiences. This is a similar approach to recommender systems.

2.3.3 Statistical Analysis

A more statistical approach is used to find patterns in EHR data on a dataset of the Danish population [17].

First we describe the dataset. The dataset which is used to apply the statistical analysis on, are EHRs collected over 15 years on over 6 million patients in Denmark. The large size of this dataset makes it possible to retrieve significant results.

We start with finding pairs of diagnoses which have a strong correlation between the diagnoses. After finding the correlated pairs, a test for directionality is applied. From this, only the pairs with a high enough correlation for a direction are kept.

The directed pairs are then connected when they have overlapping diagnoses into longer trajectories. The trajectories are then clustered. From the clusters, diagnoses can be found which are key in the disease progression. Those key diagnoses could be used to predict disease progression of patients.

The found clusters will be used to validate our approach described in chapter 4. You can find an example of a clustered trajectory in figure 2.2.

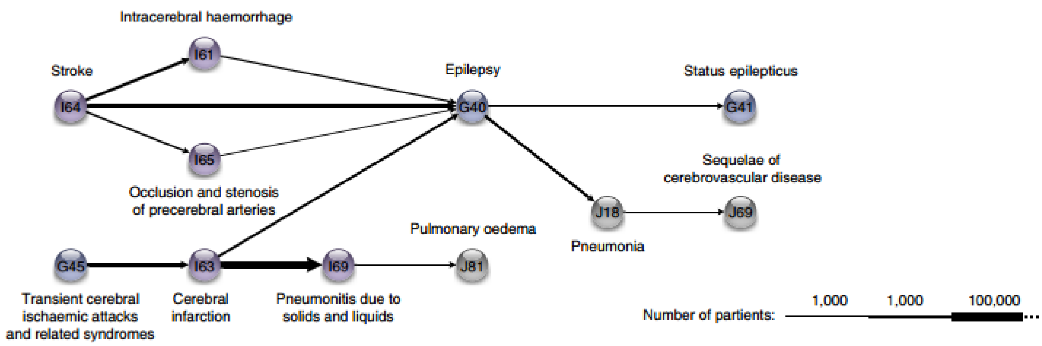


Figure 2.2: Cerebrovascular disease trajectory cluster for the Danish population [17]

2.4 Conclusion

We conclude that EHRs contain important information of a patients medical history and current state. When a large amount of data of EHRs is available, empirical results can be found in the form of patterns. Those pattern can be used to predict

and improve medical outcomes on a personal level.
The methods we describe vary from simple to very complex. But there is still room for improvement, especially in the field of advanced machine learning algorithms. The results of the Danish paper can be used to have a first validation of our approach.

In the next chapter we explain the needed background knowledge to understand the other chapters of this thesis.

Chapter 3

Background

3.1 Introduction

In this chapter we introduce important concepts which are needed to understand our approach. These concept can be used to solve problems described in chapter 2.

We start with explaining some background knowledge as time series and machine learning. We then focus on neural networks, which is a machine learning approach. The main part to understand our approach is the introduction of Word2vec. This is then used to introduce Deepwalk as an extension on Word2Vec.

3.2 Background Knowledge

3.2.1 Time Series Analysis

A time series consists of data points over a certain time period. We refer to this as a sequence of states. Where a state represents a data point and can differ from a single value to more complex representations like pictures.
The domain of time series analysis handles around extracting information or relations from a time series. It can have different goals like forecasting, classification, or exploratory.

A medical history of a patient can be seen as a time series. This means that methods applied on time series are also applicable on the medical data to find patterns.

3.2.2 Machine Learning

Machine learning is a data driven approach with as goal to build a model which can be used to make predictions or decisions. Note that this model can be used to predict outcomes of time series. This task is done by algorithms which are able to learn models based on examples given by the designer. Based on the examples, machine learning aims to tackle 3 types of problems, namely supervised learning,

unsupervised learning, and reinforcement learning.
Supervised learning is concerned with the learning task where there are examples given with their corresponding label. Unsupervised learning is similar to supervised learning only no labels are given. We won't go into reinforcement learning.
We can also classify the problems according to the desired output of our model. Those main tasks consist of classification, regression, and clustering.

We mention some used methods in the field of machine learning. These are used in all above mentioned problems.
In the field of classification neural networks are used to achieve state of the art results. For regression, Support Vector Machines can be used. One of the most popular methods for clustering is K-means.

3.2.3 Neural Networks

A neural network is a machine learning approach based on biological neural networks.

Perceptron

The basic component of a neural network is a perceptron. A perceptron takes multiple binary inputs and has a single binary output (see figure 3.1). Each input has a corresponding real numbered weight. The output is decided on the following equation:

$$output = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{ threshold} \\ 1 & \text{if if } \sum_j w_j x_j > \text{ threshold} \end{cases} \tag{3.1}$$

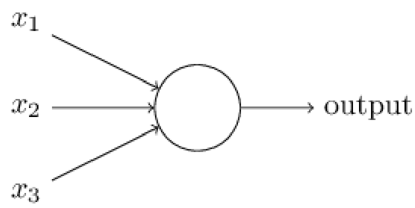


Figure 3.1: Simple presentation of a perceptron [25]

We can build a network by connecting multiple perceptrons (see figure 3.2). By building these networks, more complex decisions can be made. The reason for this, is that once there are atleast 3 layers of perceptrons, the network can find non-linear relations between the input and output.

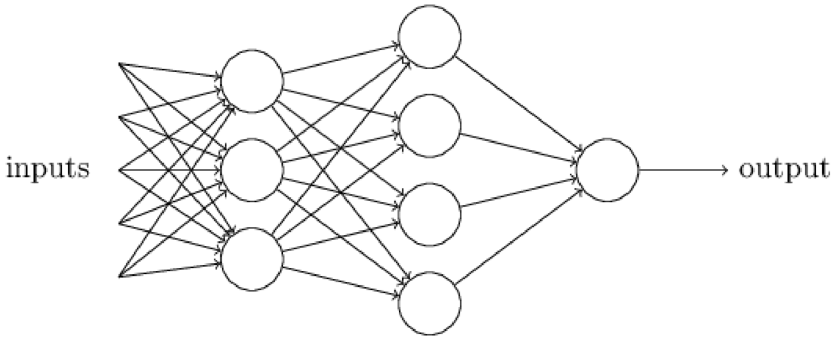


FIGURE 3.2: More complex network made by connecting multiple perceptrons [25]

Now we have seen how a general network is constructed, we look at some vocabulary.
In figure 3.3, we see a four-layer network. As mentioned on the figure, we call the first layer the input layer, the last layer the output layer, and the layers in between are called hidden layers. Sometime multiple layer networks are referred to as multilayer perceptrons or MLPs.

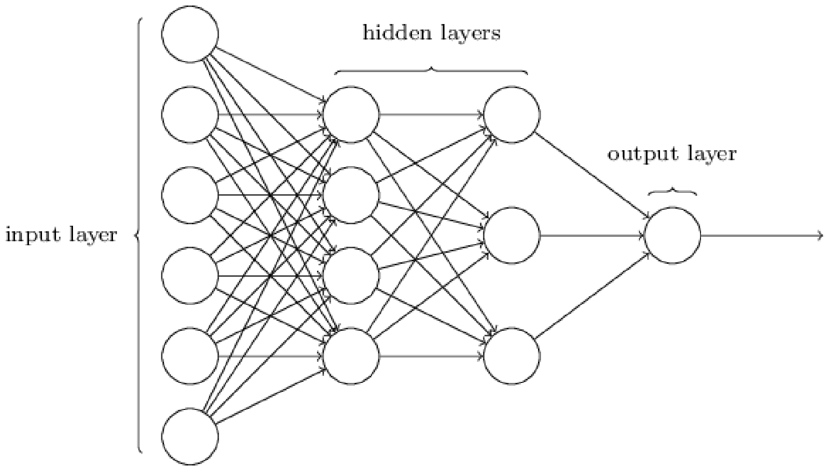


FIGURE 3.3: General vocabulary for a multilayer network [25]

Training a network

To train a neural network, we input an example with a known label. The network will calculate a certain output based on the current weights. When this output is incorrect, it should be possible to adjust the weights with as effect that the network now has as output the correct label. Note that the change in weights, should only effect the output by a small bit (see figure 3.4). The reason for this is that otherwise all the previous images could now be labeled incorrectly. So, the concept of training a neural network means, adjusting the weights in a way that the behavior of the network doesn't change completely on the previous seen pictures but that the current picture is labeled correctly.

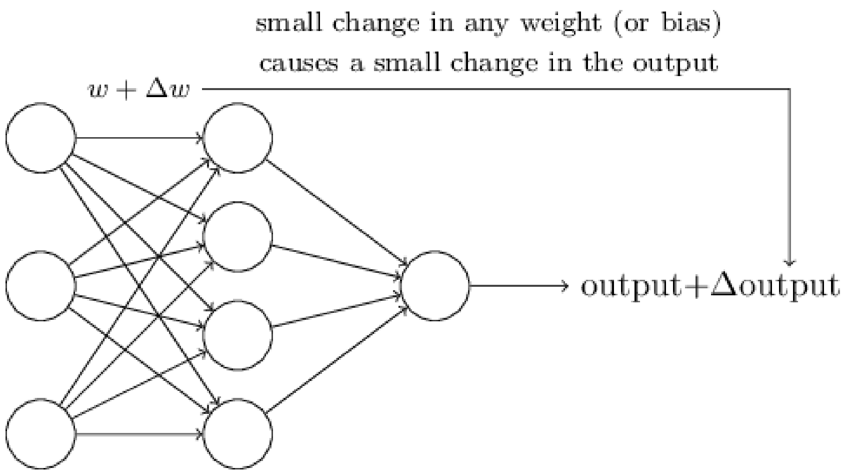


Figure 3.4: Small change on the weights, only has a small impact on the output [25]

To achieve this effect, we change our known perceptrons to sigmoid neurons. A sigmoid neuron has the same basics as a perceptron. It still has inputs but now it also has a bias b. The inputs still have weights but the weights can now range between 0 and 1. The output is now calculate with $\sigma(w * x + b)$ where σ is the sigmoid function. This results in the following formula:

$$\frac{1}{1 + exp(-\sum_j w_j x_j - b)} \tag{3.2}$$

The sigmoid function makes it possible to calculate the gradient and makes the output a linear combination of Δw_j and Δb as $\Delta output$ is approximated by

$$\Delta output \approx \sum_j \frac{\partial output}{\partial w_j} \Delta w_j + \frac{\partial output}{\partial b} \Delta b \tag{3.3}$$

Because of the linearity, it is now possible to choose changes for the weights and biases to achieve a correct output. By adjusting the weights, we will train our network to achieve a higher accuracy.

3.2.4 Backpropagation

Backpropagation is an algorithm which is used to train neural networks. It calculates the gradient of a chosen cost function with respect to the individual weights. With the gradient, the weights are updated and the cost function is minimized.

Terminology

We use w_{jk}^l to denote the weight corresponding to the connection between the k^{th} node in the $(l-1)^{th}$ layer to the j^{th} node in the l^{th} layer. We use b_j^l for the bias of the j^{th} node in the l^{th} layer and a_j^l for the activation of the j^{th} node in the l^{th} layer. See figure 3.5.

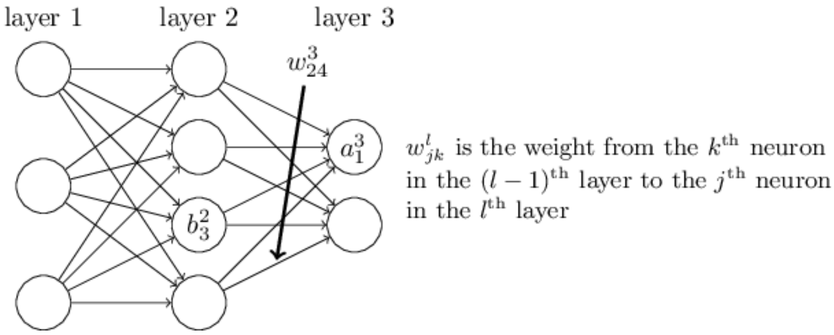


FIGURE 3.5: Visual representation of the terminology for a neural network [25]

We can now convert these notation to a vector representation. We remove the indexes for the node numbers which results in the following:

$$a^l = \sigma(w^l a^{l-1} + b^l) \tag{3.4}$$

Cost function

As mentioned before, backpropagation has as goal to calculate the partial derivatives of the cost function C with respect to each weight and bias.
The cost function has to fulfill certain criteria. The first one is that it needs to be possible to write it as a summation over cost functions for individual training examples. Secondly, it needs to be derivable. And lastly, the cost function is a function of the activations of the last layer.

Fundamental equations

Backpropagation has 4 equations. They allow us to calculate the error for each node and adjust the weights based on the gradient descent.

First we calculate the error of each node which is based on how much the cost function is influenced by each activation and on how much the activation function is influenced by z_j^L:

$$\delta_j^L = \frac{\partial C}{\partial a_j^L}\sigma'(z_j^L) \text{ with } z_j^L = \sum_k w_{jk}^L a_k^{L-1} + b_j^L \tag{3.5}$$

This can be written as a neat vector equation:

$$\delta^L = (a^L - y) \circ \sigma'(z_j^L) \tag{3.6}$$

The next equation explains why the algorithm is called backpropagation. The equation calculates each layers error vector based on the layer after it, it propagates the error back over the layers:

$$\delta^l = ((w^{l+1})^T \delta^{l+1}) \circ \sigma'(z_j^L) \tag{3.7}$$

With those 2 equations we calculate the error in each layer of the neural network. Those errors can be used to calculate the derivatives of the cost function with respect to the weights and the biases:

$$\frac{\partial C}{\partial w_{jk}^l} = a_k^{l-1} \delta_j^l. \tag{3.8}$$

$$\frac{\partial C}{\partial b_j^l} = \delta_j^l. \tag{3.9}$$

When the derivatives are calculated, we can apply the gradient descent and update the weights and biases accordingly. This process represents the learning of a neural network.

3.3 Word2Vec

3.3.1 Motivation

In natural language processing tasks, a good representation of words helps learning algorithms perform better. A representation is learned which maps words to vectors in a low-dimensional space compared to the vocabulary size. In this representation, we try to map context-similar words close to each other in the new vector space.
We could say in an informal way: a linguistic background is made which the learning algorithm can use.

3.3.2 Skip-gram

There are two main models used for word2vec [24], namely Continuous Bag-of-Words (CBOW) and the Skip-Gram model.
The first one tries to predict a word if a context is given (ex. predict Paris when capital France is given). And the second one does the inverse of this approach [29]. Empirical results have shown that the Skip-Gram model tends to do better on larger datasets [34] and gives a better representation for infrequent words [1]. In medical data there are often infrequent cases which are important. For those reasons, we choose to go further with the Skip-Gram model.

So one way of learning a word2vec representation of a corpus $Text$, is by using the skip-gram model.

Based on given words w and their contexts c, we set the parameter θ of $p(c|w;\theta)$ to maximize:

$$\arg\max_{\theta} \prod_{(w,c \in D} p(c|w;\theta) \tag{3.10}$$

with D the set of all word and context pairs we extract from the corpus. Here we also note that $p(c|w)$ is indeed the chance of a context appearing after seeing a specific word as mentioned before.

Finding word-context pairs

Given a sequence of words, we define their context based on n-gram [14]. In figure 3.6, n-gram is explained on the sentence "This is a sentence".

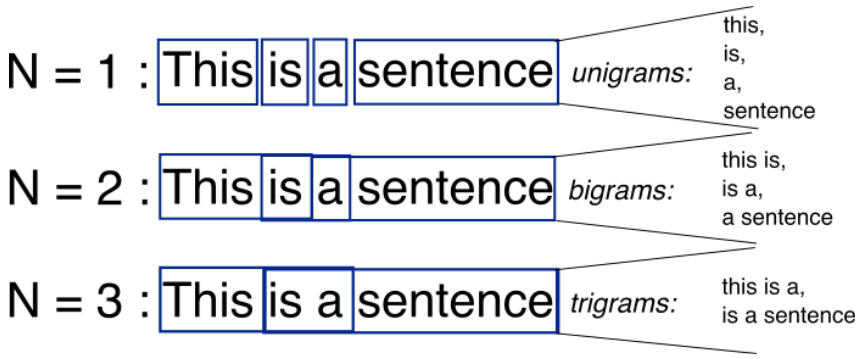


FIGURE 3.6: Explanation of n-gram [4]

For the Skip-gram model, we define the context of a word w_t as w_{t+j} with j between $-c$ and c. A larger c results in more training examples and thus can lead to a higher accuracy but will have a longer training time.

Parameterization

We start with rewriting the conditional probability using soft-max:

$$p(c|w;\theta) = \frac{e^{v_x * v_w}}{\sum_{c' \in C} e^{v_{c'} * v_w}} \tag{3.11}$$

where v_c and v_w are vector representations for c and w, and C is the set of all available contexts. This means that the parameters θ are v_{c_i} and v_{w_i}. Computing the optimal parameters is very expensive because you need to calculate this over all contexts c'. We also switch from product to sum by taking the logs:

$$\arg\max_{\theta} \sum_{(w,c) \in D} \log p(c|w;\theta) = \sum_{(w,c) \in D} (\log e^{v_c * v_w} - \log \sum_{c'} e^{v_{c'} * v_w}) \tag{3.12}$$

3.3.3 Negative Sampling

To compute the vectors using the Skip-gram model more efficiently, we introduce negative sampling [11].
Instead of calculating $\sum_{c' \in C} e^{v_{c'} * v_w}$ over all contexts, we make a set D' which consists of randomly sampled word-context pairs. With this new set, we remove the costly term $\sum_{c' \in C} e^{v_{c'} * v_w}$ and replace it with $\sum_{(w,c) \in D'} e^{v_{c'} * v_w}$.

In a less formal way: we are not making sure that if words appear in the same context, their vectors are more similar than all the other word vectors, but only of several vectors chosen randomly. This makes the Skip-gram model usable in terms of speed.

3.3.4 Neural Networks

When the word2vec algorithm is trained using the Skip-gram model, one will have a lookup table. This table contains the mapping of words to their vector representation. This lookup table can be found by training a 2-layer neural network with as goal function the function described in the previous section. The training can be done with Gradient Descent for example.
The trained 2-layer neural network can be placed in front of another neural network [26]. It will convert the words to their vector representation and feed into the next neural network. It is empirically shown that this can improve the results of the neural network by putting the lookup table in front of it. As mentioned before, in a way, you offer background knowledge to the neural network.

3.4 DeepWalk

DeepWalk is an approach where graph structured data is transformed into sequences of vertices [28]. Word2vec is then applied on those sequences to learn a good vector representation for the vertices.

In figure 3.7, we see an overview of the DeepWalk algorithm. It exist of two parts.

Algorithm 1 DEEPWALK(G, w, d, γ, t)

Input: graph $G(V, E)$
window size w
embedding size d
walks per vertex γ
walk length t
Output: matrix of vertex representations $\Phi \in \mathbb{R}^{|V| \times d}$
Initialization: Sample Φ from $\mathcal{U}^{|V| \times d}$
Build a binary Tree T from V
for $i = 0$ to γ **do**
  $\mathcal{O} = \text{Shuffle}(V)$
  for each $v_i \in \mathcal{O}$ **do**
    $\mathcal{W}_{v_i} = RandomWalk(G, v_i, t)$
    SkipGram(Φ, $\mathcal{W}_{v_i}$, w)
  end for
end for

FIGURE 3.7: Overview of the DeepWalk algorithm [28]

First a random walk generator. For each vertex v_i of the graph G, it will generate a random walk of length t. It will do this γ times but the order to which the vertices are traversed, is randomly ordered each pass. With those walks, a sequence of vertices is generated.
Secondly, those vertices are used for word2vec. This process is explained in section 3.3.

3.5 Conclusion

In this chapter we talked about general machine learning concepts and focused on Word2Vec. We conclude that Word2Vec is used to find good representation of words based on their context. It also causes that similar words will be close to each other in this new representation.

With the concepts explained in this chapter, we can introduce our approach in the next chapter. This approach is used to find patterns in EHR data.

Chapter 4

General Word2vec Approach

4.1 Introduction

In this chapter

Joins of dataset DiseaseMapping (generalization) General approach Knn KDtree Deepwalk

4.2 Conclusion

The final section of the chapter gives an overview of the important results of this chapter. This implies that the introductory chapter and the concluding chapter don't need a conclusion.

Chapter 5

Implementation

5.1 Introduction

In this chapter

OSIM Clusters TensorFlow DL4J

5.2 Dataset

To validate the approaches mentioned in chapter 4, we used a dataset generated by OSIM2. This dataset is used by OMOP to validate their methods to predict the effects of drug treatments. It contains around 10 million of hypothetical patients based on Thomson Reuters MarketScan Lab Database (MSLR). MSLR contains administrative claims between 2003 ad 2009 from a privately-insured population.

The OSIM2 dataset is contains multiple database tables which are dumped as comma-separated values (csv) files. To make it easier to work with this dataset, we joined the multiple files into one file with on each row an event of a patient containing all relevant information.

Using our approach on this dataset, we can compare our results to the found clusters in Anders Boeck Jensen et. [DANISH PAPER].

5.3 Software

5.3.1 TensorFlow

TensorFlow is a machine learning software library released at the end of 2015. It is developed by the Google Brain Team.

5.3.2 DeepLearning4Java

As mentioned in section 3.3, a trained 2-layer neural network can be placed before another neural network and function as a lookup table. In this section, we discuss a possible neural network which allows us to further investigate the effectiveness of our word2vec approach to classify patients. More concrete: we should check if a better accuracy is acquired with the lookup table in front of the neural network or without.

5.4 Conclusion

The final section of the chapter gives an overview of the important results of this chapter. This implies that the introductory chapter and the concluding chapter don't need a conclusion.

http://omop.org/OSIM2

Chapter 6

Experiments

Chapter 7

Discussion

Chapter 8

Conclusion

The final chapter contains the overall conclusion. It also contains suggestions for future work and industrial applications.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Chapter 9

Future Work

9.1 Introduction

In this chapter

Normalization Distributed RNN

9.2 Generalization

In our word2vec approach we applied generalization on the medical states. This was needed to retrieve more general n-grams. For this generalization, we divided some attributes into specific intervals.

Instead of dividing some attributes into specific intervals, we could apply normalization to it. Based on the distribution of the data, we can make more sensible intervals and assign them to the attributes.

9.3 Distributed Word2Vec

Word2vec can be made distributed as the underlying idea is quite simplistic, it counts occurrences of n-grams. Counting occurrences based on labels, is a well known problem and is often solved by MapReduce algorithms.

9.4 Patient Classification

As mentioned in section 3.3, a trained 2-layer neural network can be placed before another neural network and function as a lookup table. In this section, we discuss a possible neural network which allows us to further investigate the effectiveness of our word2vec approach to classify patients. More concrete: we should check if a better accuracy is acquired with the lookup table in front of the neural network or without.

9.4.1 Problem Definition

The medical history of a patient is seen as a time series with as datapoints an EHR. Based on the time series, we want to classify it into different disease trajectories. A patient who is classified into a specific disease trajectory, can be treated more specifically.

The medical data of multiple patients is a 3 dimensional tensor, see figure 9.1. This data structure is the input structure for a neural network.

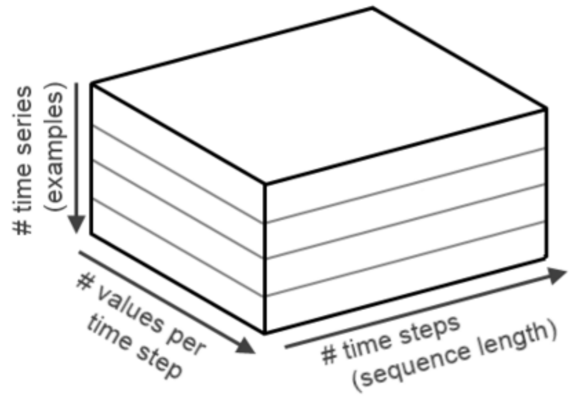


Figure 9.1: Overview of the data structure for medical data with a time aspect.

Medical data has some problems which we will discuss.
It often consists of long time periods. This means there could be a long range of dependencies between events. In the context of training neural networks, this can cause a problem known as the vanishing gradient problem.
Patients don't have regular intervals in their medical data. The irregular intervals need to be transformed to regular intervals otherwise the time aspect won't be consistent throughout the data.
The standardization of the attributes needs to be taken into account. Preferably some sort of normalization should be applied as well.
Medical data has a high dimensionality. A lot of parameters need to be taken into account to retrieve accurate results. This causes the well known problem: Curse of Dimensionality. It causes the data to be sparse and therefore, more data is needed. Especially in medical data where outliers are important.

9.4.2 Approach

Here we describe our approaches for the problems mentioned in the previous section. We solve the vanishing gradient problem with a special for of recurrent neural network, see section 9.4.3.
By applying our word2vec approach, the input is projected on another vector space using a lookup table. This vector space causes normalization. The standardization

is also done in our word2vec approach.
As we mentioned, the Curse of Dimensionality causes the need for more data. The neural network in section 9.4.3 often handles high dimensional data. In a sense, because it keeps track of the time aspect of the data, it uses the data more thoroughly and thus has a better method to handle the high dimensionality.

Padding and Masking

The transformation of the irregular interval to a regular one, is done with padding and masking.

If we don't use any masking and padding, our data can only be of equal length time series and at each time step a classification. Our data consists of several inputs, the different time steps of a time series, and has one output associated with it, namely the classification of the time series.

The method of padding is simply by adding empty events (ex. zeros) to the shorter time series until all examples are of equal length for both input and output. Using padding, changes the data quite drastically and would case problems during the training because of that. For this problem we use the method of masking. With masking, we have two additional arrays which contain the information about whether an input or output is padding or not. See figure 9.2, picture 2 on how the masking is done for a many to one case.

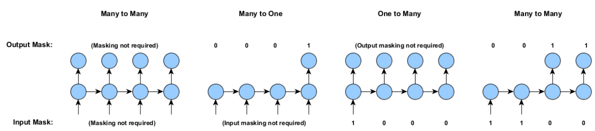


Figure 9.2: Multiple masking methods.

9.4.3 Neural Network

We mentioned the reasons on why we choose a Long Short Term Memory (LSTM) approach as a recurrent neural network solution. In this section we explain in more detail why LSTM handles the vanishing problem and long-term dependencies.

First we shortly repeat the structure of a neural network in figure 9.3. We see the input, different layers with their perceptrons, and σ as the activation function.

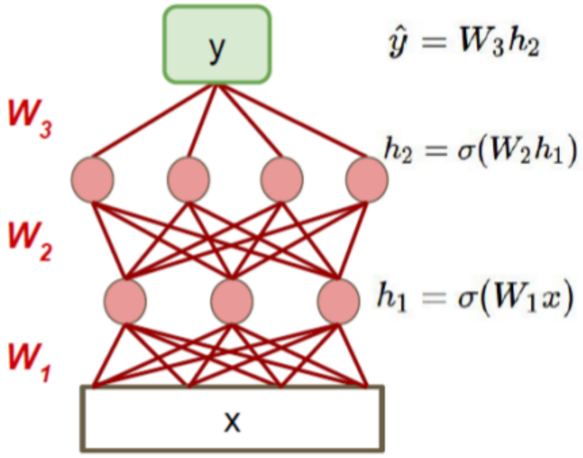


Figure 9.3: A general structure of a neural network.

Recurrent Neural Network

A standard neural network don't have any persistence. They will classify their input but when they get a stream of inputs (ex. speech), they will classify each word independently of each other and without any regards of the previous words. A recurrent neural network (RNN) addresses this problem by introducing networks with loops. This way, the output of a previous input has effect on the next input. In figure 9.4, we transform those loops into multiple copies of the same network which makes it easier to reason about.

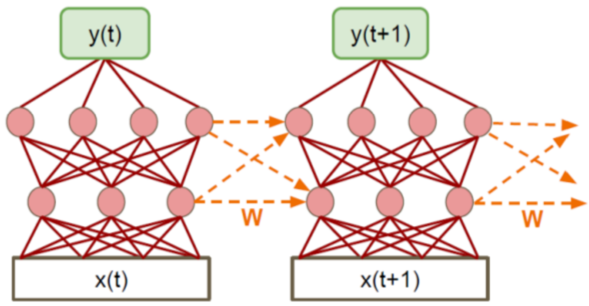


Figure 9.4: An unrolled recurrent neural network.

The problem with RNNs is mainly that they have troubles learning long-term dependencies which is often essential in time series.

Long Short Term Memory

A LSMT network is a specific RNN which is capable of learning long-term dependencies. We will explain the difference with a standard RNN and why a LSTM can learn these long-term dependencies.

A recurrent network is, as we said, a chain of connected neural networks. Those networks can have a simple structure as a single *tanh* layer, see figure 9.5.

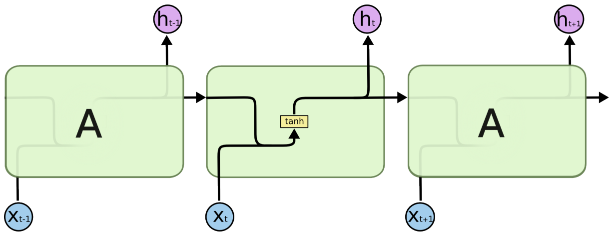


FIGURE 9.5: An unrolled recurrent neural network with a single tanh layer.

It is important to see the difference with a LSTM. The repeating network doesn't have a single neural network layer, but has 4 layers which each fulfills a specific goal, see figure 9.6.

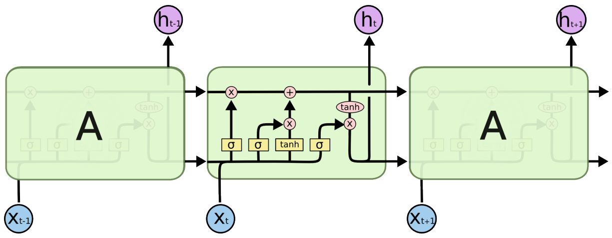


FIGURE 9.6: An unrolled LSTM network where each network has 4 layers.

The main idea behind LSTM is that each repeated network has its own cell state. It functions as a memory which can be updated with each new input. On figure 9.7, you can see the cell state C through time. It can be compared with a conveyor belt which interacts with the input at certain gates. This way the state is updated throughout several inputs.

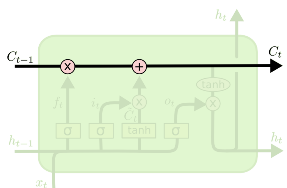
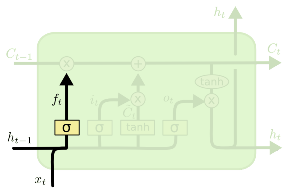


FIGURE 9.7: Representation of the cell state for a LSTM network.

In the following figures, we show the different gates and their functions in changing the cell state depending on the input and the output of the previous network. Next to each figure the formulas are shown on how the cell state is updated. There should be no surprises as they are not much different than the standard formulas of neural networks.

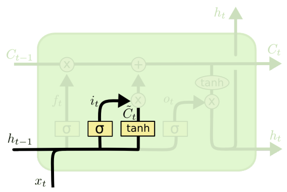
We start with the forget gate layer of a LSTM. Based on x_t and h_{t-1}, it outputs a number between 0 and 1 for each number in the cell state C_{t-1}. This is shown in figure 9.8.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

FIGURE 9.8: The forget layer of a LSTM network.

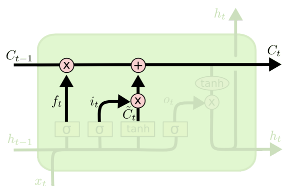
Next we look to the input gate layer. This gate decides which values will be updated in the cell state and outputs those in i_t. It is then combined with the vector $\widetilde{C}_t$, which contains the new candidate values based on the input x_t and h_{t-1}. This is shown in figure 9.9.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \ + \ b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \ + \ b_C)$$

FIGURE 9.9: The input layer of a LSTM network.

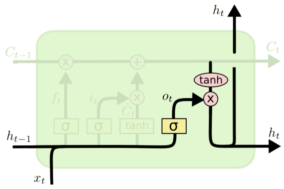
We can now combine the previous results and adjust the cell state. We multiply the old state with f_t so we forget the needed elements. Then we add $i_t * \widetilde{C}_t$ which are the new candidate values multiplied by the amount on how much we want to update each state value. See figure 9.10.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

FIGURE 9.10: Update process of the cell state of a LSTM network.

Finally, we need to output h_t to the next network. This is based on the input and the cell state C_t. See figure 9.11.



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

FIGURE 9.11: Decide the output of a LSTM network.

Variants Long Short Term Memory

In [LSTM: SEARCH SPACE ODESEY], research is done between different variant of LSTM networks. It was concluded that the forget gate and the activation function is the most important. Other variants don't have a large influence and mainly add a lot of extra complexity.

http://www-dsi.ing.unifi.it/p̃aolo/ps/tnn-94-gradient.pdf

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

http://deeplearning4j.org/usingrnns

Supervised Sequence Labelling with Recurrent Neural Networks

Phenotyping of Clinical Time Series with LSTM Recurrent Neural Networks

IMEC Technical talk by Jaak Simm

Missing covariate data in medical research: To impute is better than to ignore

Development of a Database of Health Insurance Claims: Standardization of Disease Classifications and Anonymous Record Linkage

Modeling Temporal Dependencies in High- Dimensional Sequences: Application to Polyphonic Music Generation and Transcription

Long short-term memory neural network for traffic speed prediction using remote microwave sensor data

Noisy Time Series Prediction using a Recurrent Neural Network and Grammatical Inference

http://colah.github.io/posts/2015-08-Understanding- LSTMs

A Critical Review of Recurrent Neural Networks for Sequence Learning

LONG SHORT-TERM MEMORY

LSTM: A Search Space Odyssey

9.5 Conclusion

The final section of the chapter gives an overview of the important results of this chapter. This implies that the introductory chapter and the concluding chapter don't need a conclusion.

Appendices

Appendix A

The First Appendix

Appendices hold useful data which is not essential to understand the work done in the master thesis. An example is a (program) source. An appendix can also have sections as well as figures and references[**?**].

A.1 More Lorem

Quisque facilisis auctor sapien. Pellentesque gravida hendrerit lectus. Mauris rutrum sodales sapien. Fusce hendrerit sem vel lorem. Integer pellentesque massa vel augue. Integer elit tortor, feugiat quis, sagittis et, ornare non, lacus. Vestibulum posuere pellentesque eros. Quisque venenatis ipsum dictum nulla. Aliquam quis quam non metus eleifend interdum. Nam eget sapien ac mauris malesuada adipiscing. Etiam eleifend neque sed quam. Nulla facilisi. Proin a ligula. Sed id dui eu nibh egestas tincidunt. Suspendisse arcu.

A.1.1 Lorem 15–17

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi.

In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.

A.1.2 Lorem 18–19

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

A.2 Lorem 51

Maecenas dui. Aliquam volutpat auctor lorem. Cras placerat est vitae lectus. Curabitur massa lectus, rutrum euismod, dignissim ut, dapibus a, odio. Ut eros erat, vulputate ut, interdum non, porta eu, erat. Cras fermentum, felis in porta congue, velit leo facilisis odio, vitae consectetuer lorem quam vitae orci. Sed ultrices, pede eu placerat auctor, ante ligula rutrum tellus, vel posuere nibh lacus nec nibh. Maecenas laoreet dolor at enim. Donec molestie dolor nec metus. Vestibulum libero. Sed quis erat. Sed tristique. Duis pede leo, fermentum quis, consectetuer eget, vulputate sit amet, erat.

Appendix B

The Last Appendix

Appendices are numbered with letters, but the sections and subsections use arabic numerals, as can be seen below.

B.1 Lorem 20-24

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus commodo lacus. Morbi lacus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac lacus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis orci. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetuer quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, suscipit in, pharetra placerat, metus. Vivamus commodo dui non odio. Donec et felis.

Etiam suscipit aliquam arcu. Aliquam sit amet est ac purus bibendum congue. Sed in eros. Morbi non orci. Pellentesque mattis lacinia elit. Fusce molestie velit in ligula. Nullam et orci vitae nibh vulputate auctor. Aliquam eget purus. Nulla auctor wisi sed ipsum. Morbi porttitor tellus ac enim. Fusce ornare. Proin ipsum enim, tincidunt in, ornare venenatis, molestie a, augue. Donec vel pede in lacus sagittis porta. Sed hendrerit ipsum quis nisl. Suspendisse quis massa ac nibh pretium cursus. Sed sodales. Nam eu neque quis pede dignissim ornare. Maecenas eu purus ac urna tincidunt congue.

Donec et nisl id sapien blandit mattis. Aenean dictum odio sit amet risus. Morbi purus. Nulla a est sit amet purus venenatis iaculis. Vivamus viverra purus vel magna. Donec in justo sed odio malesuada dapibus. Nunc ultrices aliquam nunc. Vivamus facilisis pellentesque velit. Nulla nunc velit, vulputate dapibus, vulputate id, mattis ac, justo. Nam mattis elit dapibus purus. Quisque enim risus, congue non, elementum ut, mattis quis, sem. Quisque elit.

Maecenas non massa. Vestibulum pharetra nulla at lorem. Duis quis quam id lacus dapibus interdum. Nulla lorem. Donec ut ante quis dolor bibendum condimentum. Etiam egestas tortor vitae lacus. Praesent cursus. Mauris bibendum pede at elit. Morbi et felis a lectus interdum facilisis. Sed suscipit gravida turpis. Nulla at lectus. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Praesent nonummy luctus nibh. Proin turpis nunc, congue eu, egestas ut, fringilla at, tellus. In hac habitasse platea dictumst.

B.2 Lorem 25-27

Vivamus eu tellus sed tellus consequat suscipit. Nam orci orci, malesuada id, gravida nec, ultricies vitae, erat. Donec risus turpis, luctus sit amet, interdum quis, porta sed, ipsum. Suspendisse condimentum, tortor at egestas posuere, neque metus tempor orci, et tincidunt urna nunc a purus. Sed facilisis blandit tellus. Nunc risus sem, suscipit nec, eleifend quis, cursus quis, libero. Curabitur et dolor. Sed vitae sem. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Maecenas ante. Duis ullamcorper enim. Donec tristique enim eu leo. Nullam molestie elit eu dolor. Nullam bibendum, turpis vitae tristique gravida, quam sapien tempor lectus, quis pretium tellus purus ac quam. Nulla facilisi.

Duis aliquet dui in est. Donec eget est. Nunc lectus odio, varius at, fermentum in, accumsan non, enim. Aliquam erat volutpat. Proin sit amet nulla ut eros consectetuer cursus. Phasellus dapibus aliquam justo. Nunc laoreet. Donec consequat placerat magna. Duis pretium tincidunt justo. Sed sollicitudin vestibulum quam. Nam quis ligula. Vivamus at metus. Etiam imperdiet imperdiet pede. Aenean turpis. Fusce augue velit, scelerisque sollicitudin, dictum vitae, tempor et, pede. Donec wisi sapien, feugiat in, fermentum ut, sollicitudin adipiscing, metus.

Donec vel nibh ut felis consectetuer laoreet. Donec pede. Sed id quam id wisi laoreet suscipit. Nulla lectus dolor, aliquam ac, fringilla eget, mollis ut, orci. In pellentesque justo in ligula. Maecenas turpis. Donec eleifend leo at felis tincidunt consequat. Aenean turpis metus, malesuada sed, condimentum sit amet, auctor a, wisi. Pellentesque sapien elit, bibendum ac, posuere et, congue eu, felis. Vestibulum mattis libero quis metus scelerisque ultrices. Sed purus.

Bibliography

[1] Google code archive - long-term storage for google code project hosting. https://code.google.com/archive/p/word2vec/. (Accessed on 05/16/2016).

[2] Healthit.gov | the official site for health it information. https://www.healthit.gov/. (Accessed on 05/15/2016).

[3] Meddra. http://www.meddra.org/. (Accessed on 05/03/2016).

[4] The speech recognition wiki. http://recognize-speech.com/. (Accessed on 05/16/2016).

[5] WHO | International Classification of Diseases (ICD). http://www.who.int/classifications/icd/en/. (Accessed on 04/27/2016).

[6] Y. Bengio, P. Simard, and P. Frasconi. Learning Long-term Dependencies with Gradient Descent is Difficult. *Trans. Neur. Netw.*, 5(2):157–166, Mar. 1994.

[7] C. Bennett and T. Doub. Data Mining and Electronic Health Records: Selecting Optimal Clinical Treatments in Practice. *CoRR*, abs/1112.1668, 2011.

[8] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent. Modeling Temporal Dependencies in High-Dimensional Sequences: Application to Polyphonic Music Generation and Transcription. *ArXiv e-prints*, June 2012.

[9] A. G. Chris Nicholson. Using Recurrent Neural Networks in DL4J - Deeplearning4j: Open-source, distributed deep learning for the JVM. http://deeplearning4j.org/usingrnns. (Accessed on 05/03/2016).

[10] C. L. Giles, S. Lawrence, and A. C. Tsoi. Noisy time series prediction using recurrent neural networks and grammatical inference. *Mach. Learn.*, 44(1-2):161–183, July 2001.

[11] Y. Goldberg and O. Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722, 2014.

[12] A. Graves. *Supervised Sequence Labelling with Recurrent Neural Networks*, volume 385 of *Studies in Computational Intelligence*. Springer, 2012.

[13] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber. LSTM: A Search Space Odyssey. *CoRR*, abs/1503.04069, 2015.

[14] D. Guthrie, B. Allison, W. Liu, L. Guthrie, and Y. Wilks. A Closer Look at Skip-gram Modelling.

[15] A. Hameurlain, J. Kung, R. Wagner, H. Decker, L. Lhotska, and S. Link, editors. *Transactions on Large-Scale Data- and Knowledge-Centered Systems IV - Special Issue on Database Systems for Biomedical Applications*, volume 8980 of *Lecture Notes in Computer Science*. Springer, 2011.

[16] K. J. M. Janssen, A. R. T. Donders, F. E. J. Harrell, Y. Vergouwe, Q. Chen, D. E. Grobbee, and K. G. M. Moons. Missing covariate data in medical research: To impute is better than to ignore. *Journal of Clinical Epidemiology*, 63(7):721–727, 2016.

[17] A. B. Jensen, P. L. Moseley, T. I. Oprea, S. G. Ellesoe, R. Eriksson, H. Schmock, P. B. Jensen, L. J. Jensen, and S. Brunak. Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients. *Nat Commun*, 5, Jun 2014. Article.

[18] C. K. R. Jimeng Sun. Big Data Analytics for Healthcare. 2013.

[19] A. Karpathy. The Unreasonable Effectiveness of Recurrent Neural Networks. http://karpathy.github.io/2015/05/21/rnn-effectiveness/. (Accessed on 05/03/2016).

[20] S. Kimura, T. Sato, S. Ikeda, M. Noda, and T. Nakayama. Development of a database of health insurance claims: Standardization of disease classifications and anonymous record linkage. *J Epidemiol*, 20(5):413–419, Sep 2010. 20699602[pmid].

[21] Z. C. Lipton. A critical review of recurrent neural networks for sequence learning. *CoRR*, abs/1506.00019, 2015.

[22] Z. C. Lipton, D. C. Kale, and R. C. Wetzel. Phenotyping of clinical time series with LSTM recurrent neural networks. *CoRR*, abs/1510.07641, 2015.

[23] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies*, 54:187 – 197, 2015.

[24] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. *CoRR*, abs/1310.4546, 2013.

[25] M. Nielsen. Neural networks and deep learning. http://neuralnetworksanddeeplearning.com/. (Accessed on 05/03/2016).

[26] C. Olah. Deep Learning, NLP, and Representations - colah's blog. http://colah.github.io/posts/2014-07-NLP-RNNs-Representations/. (Accessed on 05/16/2016).

[27] C. Olah. Understanding lstm networks – colah's blog. http://colah.github.io/posts/2015-08-Understanding-LSTMs/. (Accessed on 05/03/2016).

[28] B. Perozzi, R. Al-Rfou, and S. Skiena. DeepWalk: Online Learning of Social Representations. *CoRR*, abs/1403.6652, 2014.

[29] X. Rong. word2vec parameter learning explained. *CoRR*, abs/1411.2738, 2014.

[30] H. Sak, A. W. Senior, and F. Beaufays. Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *CoRR*, abs/1402.1128, 2014.

[31] J. Simm. IMEC Technical talk.

[32] C. P. Stone. A Glimpse at EHR Implementation Around the World: The Lessons the US Can Learn. 2014.

[33] J. Sun, F. Wang, J. Hu, and S. Edabollahi. Supervised patient similarity measure of heterogeneous patient records. *SIGKDD Explor. Newsl.*, 14(1):16–24, Dec. 2012.

[34] G. B. Team. Vector representations of words. https://www.tensorflow.org/versions/0.6.0/tutorials/word2vec/index.html. (Accessed on 05/16/2016).

[35] F. Wang, N. Lee, J. Hu, J. Sun, and S. Ebadollahi. Towards Heterogeneous Temporal Clinical Event Pattern Discovery: A Convolutional Approach. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 453–461, New York, NY, USA, 2012. ACM.

Master thesis filing card

Student: Milan van der Meer

Title: Learning a Disease Embedding using Generalized Word2Vec Approaches.

UDC: 621.3

Abstract:

Here comes a very short abstract, containing no more than 500 words. LaTeX commands can be used here. Blank lines (or the command `\par`) are not allowed!

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Thesis submitted for the degree of Master of Science in Engineering: Computer Science, specialisation Artificial Intelligence

Thesis supervisor: Prof. dr. Roel Wuyts

Assessors: Ir. Kn. Owsmuch
K. Nowsrest

Mentors: Ir. An Assistent
S. Dhooghe