Context in Information Retrieval

A Case Study with the COVID-19 Pandemic

A thesis submitted by
Marcel Braasch
for the degree of
Bachelor of Science in Computer Science
supervised by
Professor Dr. Visvanathan Ramesh

Fachbereich 12 Mathematik und Informatik
Johann Wolfgang Goethe-Universität Frankfurt am Main

September 30th, 2020

Abstract

The COVID-19 pandemic is keeping the world in suspense. A previously unseen scenario brings difficult times. Making the right decisions in times of such uncertainty requires a thorough evaluation of the information situation. Enormous amounts of data that are constantly growing make this endeavor no easier task. Overlooking the dynamic flow of information and understanding when facts, opinions and points of view change are impossible to manage manually by humans. Since their first appearance, information retrieval systems have helped to keep an overview of complex circumstances like these. The information retrieval system presented in this thesis is designed to help users to present customized results without their explicit assistance. The system uses neural methods to automatically determine the user's environment and their implicit need. As in use, the system continuously learns the user's context. Personalized results are delivered promptly and the user's information situation is updated again.

At first, important fundamentals of deep learning, information retrieval and context are presented. Thereafter, experiments using the system described above are conducted. These experiments are further evaluated, visualized and interpreted. Finally, an outlook for possible further research based on the system is given.

Acknowledgments

I would like to acknowledge everyone who has directly or indirectly helped me finish this thesis. Though it is *just* a bachelor's thesis, it forms an important milestone in my life. I struggled to find a passion for many years and finally came across computer science. Most important for this development were my parents Emeli and Jens. I want to thank you from the bottom of my heart for the patience you have had and the unconditional support you showed me at all times.

I also want to acknowledge my partner in crime Cecile. Thank you for always supporting me. You have my back, and I know I can always count on you. I am infinitely grateful to know you by my side.

My personal development in the last months can only be attributed to Professor Ramesh. I am incredibly grateful for the opportunity you have given me. The level of support, the time and the advice you contributed to me are by no means self-evident. Our Skype calls were always a highlight for me, and your view on the world simply amazes me. You pushed me to explore domains I didn't even know existed. You have truly broadened my horizon - the pages in this thesis embody only a fraction of what you taught me. I thank you from the bottom of my heart for your support, and I sincerely hope our paths cross again. All the best to you!

Lastly, special thanks go out to Mike and Henri. Thank you two for proofreading this thesis. Your feedback was really valuable. Henri, thank you for always being open to discussions when I needed someone to talk.

Contents

List of Abbreviations

ANN	Artificial neural network
CSG	Continuous skip gram model
COVID-19	Coronavirus disease 2019
DNN	Deep neural network
FFNN	Feed forward neural network
IR	Information retrieval
LSTM	Long short-term memory
NLP	Natural language processing
NMT	Neural machine translation
NN	Neural network
OA	Overall average
PCA	Principal component analysis
RNN	Recurrent neural network
SA	Single average
SBERT	Sentence-BERT
VSM	Vector space model

1 Introduction

The severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) is rapidly spreading around the globe, causing the coronavirus disease 2019 (COVID-19) pandemic. Humanity is facing a highly complex and unseen situation changing the everyday life of millions. The pandemic has to date lead to over 1.000.000 deaths. Ongoing local outbreaks and a heterogeneous severity landscape keeps the world in suspense. Rapid infection dynamics demand everyone to make instantaneous decisions weighing up various interests. Precisely because of this dynamic, the information situation is in constant motion. It is hard to keep up with different sources, opinions and the change evolving around them. Questions like 'Is a child immune to a coronavirus infection?' have had many different, presumably correct, answers in the last couple of months. Understanding which is the current school of thought and assembling the right information is difficult.

At the beginning of the pandemic, media reporting was almost exclusively occupied by COVID-19 related articles. The exact number is unknown, but the number of published research papers gives a feeling for the incredible amount of information available. Since March 2020 about 230.000 scientific papers have been published, confirming the immense thematic interest in the population (Wang et al., 2020).

Providing a system which helps to look for COVID-19 related information is a reasonable step which has been recognized by many. Examples would be scientific search engines like vespa.ai or news watches like oecd.ai. These systems, however, yield significant limitations. The first is not suitable for everyday use as it searches scientific papers only. The second indeed condenses daily news. Unfortunately, news presented are very unstructured and do not adapt to the individual's need. This work tries to over overcome these issues. It explores a promising approach to search corpora of COVID-19 related newspaper articles. A system aimed at personalizing content is presented, followed by various experiments showing the system's effectiveness.

1.1 Motivation

The COVID-19 related state of information has taken on extreme dimensions. Enormous masses of newspaper articles have already been published, and are posted every day. For humans, it is merely impossible to scan all this information and manually filter it. In most cases, two scenarios arise. Either one wants to read horizontally, covering a broad spectrum of topics. Or one wants to acquire knowledge vertically, looking for a specific topic. If the latter occurs, usually one is interested in answering a particular set of questions concerning this topic. The vertical research approach is much more focused, which is why finding perfectly fitting information is not a trivial task.

Consider the simple scenario of a manager caring for his workplace. Assume he wants to know whether his employees should work from home. He supervises about 100 employees in an open-space office. Of course, he will consult one of the available search engines (e.g., Google). Entering his query, he types

coronavirus transmission

which yields a major issue (we neglect the ambiguity of words at this point). The first word describing his information need is *transmission*. Of course, viral transmissibility properties are critical in understanding the risks of a pathogen. But more general topics such as an integrated hygiene concept, hand washing, social distancing or at what size working from home makes sense are certainly important as well. Though the need was explcitily formulated it is imprecise. The manager will quickly notice that the search results may not be sufficient and adapt his query and search style. Inaccuracies, however, will always occur if one sticks to words only.

By this simple example, it becomes evident that users may not be able to express certain information needs adequately. Even if the manager had chosen different words (or added *office* to the query), there would have been a discrepancy between his information need and the explicit search query. Humans create a cognitive representation of problems in their mind (Jonassen, 2003). As the mystery of human cognition remains an unsolved task until today, mapping this representation solely to a sequence of words will most likely always result in a loss of information.

Utilizing meaningful representations of the user's intent might help to understand the information need. Augmenting the explicit query string by this implicitly acquired knowledge will be the overall objective of this work.

1.2 Aim

This thesis aims to explore different current approaches given by the information retrieval (IR) landscape. We finally determine a promising approach, namely neural vector space models, to solve the complex problem of personalized search in the context of the COVID-19 pandemic. Readers interested in IR and neural methods are the target audience for this work. While the experiments conducted are not as extended as we wish they were, they form a possible starting point for further research in this field. The aim of this thesis is of rather theoretical nature. Due to time constraints, a reusable API as such is not provided, but the code for reproducing experiments will be made publicly available [1].

Low-level objectives for this thesis are aligned with the chapter structure. Section 2 provides a relatively inexperienced reader with a brief introduction to the fundamentals of deep learning. We begin with the basic concepts of neural networks, proceed with the training of such and introduce recent advances in the field. Throughout the entire thesis, formal mathematics is in use, requiring a basic mathematical education, especially in linear algebra and multivariate calculus. Section 3 broadly introduces the concept of IR and quickly moves on to its neural methods. The two main concepts of this work, vector space models and word embeddings, are thoroughly discussed. Section 4 introduces the idea of context in IR. Section 4.2, specifically, forms the rationale for the contexts this work makes uses. The context model introduced will finally be used in the

[1] github.com/marcelbra

last section. Section 5 embeds the context into an IR system and attempts to get an understanding of the user's surroundings. The system tries to implicitly optimize the context of the user to return the best possible search results. Experiments, simple examples and limitations are presented subsequently.

1.3 Methods

The approach shown in this thesis makes use of vector space models (see Section 3.4). The vector representations for this model are based on the BERT model (see Section 3.7). By means of similarity, the model retrieves documents closest to the query (see Section 3.3). Context relevant to the current situation of the pandemic is determined (see Section 4) and eventually used to re-rank documents. If one is interested in more details, Section 5.1 provides a much more detailed description of the system in use.

2 Fundamentals of Deep Learning

Most technological achievements of humankind are inspired by nature. In the scientific literature, this process is often referred to as bionics. Airplanes were developed and optimized under the observation of large birds. Tire treads have copied their characteristics from cat paws. Even trivial mechanisms such as hook-and-loop fasteners can be found in their original form in plants. Important achievements in modelling the world was, and still is, inspired by the structure of the human brain. The neuron structure of the brain provides new promising approaches for the development of so-called artificial neural networks (ANN).

In general, a neural network is a network of neurons transmitting impulses. Propagation happens until the corresponding region of interest is reached. ANNs are designed to simulate this impulse transmitting. Inspired by this, deep neural networks (DNN) form modelling approach for solving challenging tasks, increasingly even outperforming humans.

A feed forward neural network (FFNN), in essence, is a mathematical function (later often referred to as a *model*) mapping an arbitrary input to the desired output. These functions may be simple linear functions with very few parameters. More sophisticated approaches, however, may comprise models with millions of parameters (Devlin, Chang, Lee, & Toutanova, 2018). Artificial neural networks make use of formal mathematics, and the data they process is always in the form of vector representations. We assume that the reader is familiar with the basics of linear algebra and multivariable calculus. A foundation in vectors, matrices and their differential and non-differential operations are a prerequisite for understanding the following sections.

The following sections give the reader a broad overview of the fundamentals of current deep learning approaches. Examples of these approaches are applied to natural language processing (NLP). Not only do we want to create a notion of intuition. In our view, a real understanding of complex mathematics is only acquired by introducing formal definitions followed by practical examples. We begin with a general introduction to FFNN, explain how they compute predictions and show methods of retrieving model parameters. Building upon that, essential concepts like recurrent neural networks (RNN) and encode-decoder models are introduced. These models form predecessors of today's state-of-the-art architecture, namely transformers.

If not stated otherwise, all ideas and equations for Section 2.1 and 2.2 are taken from Goodfellow, Bengio, and Courville (2016)'s book *Deep Learning*. If not stated otherwise, all notations, illustrations, examples and explanations for the entire section are our own work.

2.1 Feed Forward Neural Networks

For the sake of simplicity a FFNN with only one hidden layer is presented in the following. An extension with more hidden layers analogous to the example provided is easy to construct. Besides, a fine-grained derivation of the matrix notation is omitted.

Definition A FFNN with one hidden layer is a function $f : \mathbb{R}^n \to \mathbb{R}^m$, such that for data pairs $\{(x_i, y_i)\}_{i=1}^n$ a (linear or non-linear) relationship is given as $f(x_i) \approx y_i$.

Forward Propagation An input $x \in \mathbb{R}^{n\times 1}$ is multiplied by a weight matrix $W_{hx} \in \mathbb{R}^{k\times n}$ and projected onto a hidden layer $h \in \mathbb{R}^{k\times 1}$. Together with a bias b_h, an activation function a_h maps the hidden layer to a new subspace. Multiplying the hidden layer by another weight matrix $W_{\hat{y}h} \in \mathbb{R}^{m\times k}$ produces the output $z \in \mathbb{R}^{m\times 1}$. Again, with a bias $b_{\hat{y}}$, an activation function $a_{\hat{y}}$ of h and b return the final result $\hat{y}$. Formally, this can be expressed as

$$\begin{aligned} \hat{y} &= a_{\hat{y}}(W_{\hat{y}h}h + b_{\hat{y}}) \\ \text{where } h &= a_h(W_{hx}x + b_h) \end{aligned} \tag{1}$$

An illustration of the above equation can be in found in Figure 1.

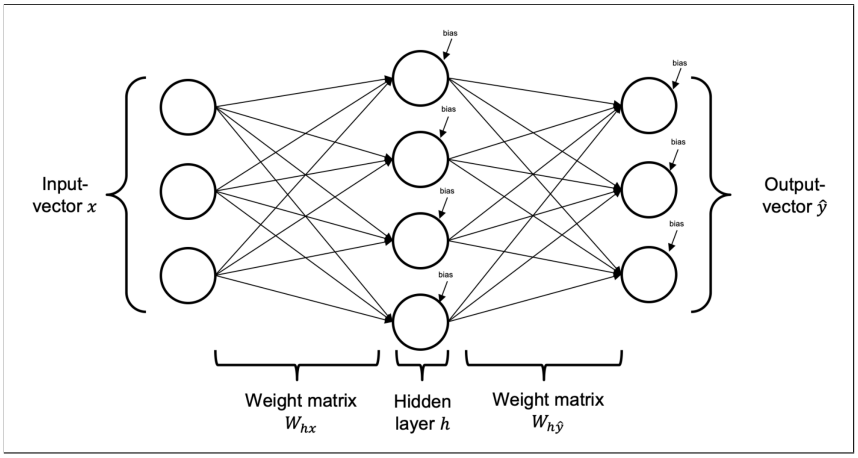

Figure 1: FFNN with one hidden layer. $n = 3$, $k = 4$ and $m = 3$. The fully connectedness illustrated is expressed by the weight matrices W_{hx} and $W_{h\hat{y}}$.

Note that, even though illustrated as *layer times matrix* it is a convention to multiply *matrix times layer*. Therefore the subscripts are, seemingly wrong, expressed in this way as well. The activation functions a are often non-linear functions such as *softmax*, *ReLU* or the *sigmoid* fucntion. This is important, because if non-linearities were not added, the network would again, just be a linear function of its arguments. It would not be able to express more complicated relationships.

Computational Graphs In order to understand the subsequent sections better, we briefly introduce computational graphs. Computational graphs can express complicated mathematical functions in a considerably simple manner. Besides, they yield the ability to perform derivations of any type of function in a

very straightforward way. For the explanation of RNNs they are also of significant importance.

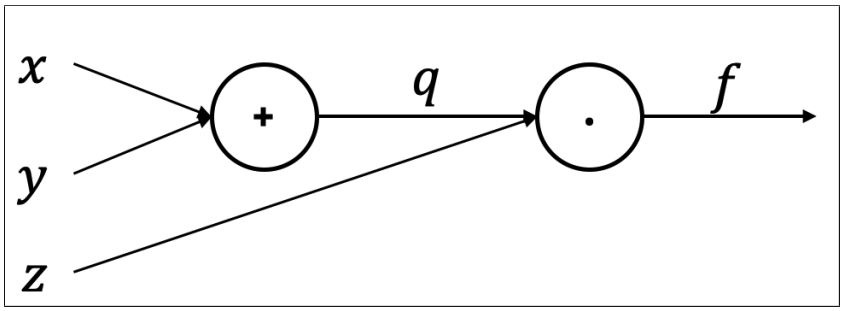

Figure 2: Computational graph of $f(x, y, z) = (x+y) \cdot z$. Example graph after Li et al. (2017).

Consider the function $f(x, y, z) = (x + y) \cdot z = q \cdot z$. An illustration of f's computational graph can be found in Figure 2. Our goal is to find the derivative of f with respect to x, y and z. In other words, we want to find out how sensitive f is to a change in each of its variables. This sensitivity, later, will be used to find the direction of a function's steepest descent. To achieve this, we apply the chain rule in a backward manner. The chain rule states, that if a variable z depends on y which depends on x, then z depends on x via the mediator y. Formally, in Leibniz's notation, this can be expressed as

$$\frac{dz}{dx} = \frac{dz}{dy} \cdot \frac{dy}{dx}. \tag{2}$$

Assume that we only want to find the change of x to f. By the chain rule, this can now be expressed as

$$\begin{aligned} \frac{df}{dx} &= \frac{df}{dq} \cdot \frac{dq}{dx} \\ &= \frac{d(q \cdot z)}{dq} \cdot \frac{d(x + y)}{dx} \\ &= z \cdot 1 \\ &= z \end{aligned}$$

Of course, the derivation could have been achieved by hand very easily. But as soon as the function becomes slightly more complicated, unrolling functions to computational graphs simplifies calculations drastically.

Backpropagation So far, we only showed how to compute an approximation $\hat{y}$ given some input data. This approximation $\hat{y}$ is supposed to be close to the real value y. Where the parameters of the FFNN come from has not yet been discussed. Given a set of training examples $\{(x_i, y_i)\}_{i=1}^{n}$ our objective is to

minimize the discrepancy between an estimated value $\hat{y}_i$ and its corresponding ground truth y_i. This objective can be expressed as a cost function to minimize. A simple example of such functions is the mean-squared-error, that is

$$L = \frac{1}{n} \sum_{i}^{n} (\hat{y}_i - y_i)^2. \tag{3}$$

Informally, we aim to move parameters towards the direction of the steepest descent in L. Minimizing L results in a low discrepancy between $\hat{y}$ and y. Formally, the gradient of L is

$$\nabla L = \left[\frac{dL}{dW_{hx}}, \frac{dL}{dW_{\hat{y}h}}, \frac{dL}{db_h}, \frac{dL}{db_{\hat{y}}} \right]. \tag{4}$$

By plugging in Equation 1 into L one can quickly see how calculating the derivative for a presumably simple one-layer neural network becomes a hard task very quickly. Computational graphs can now be used to calculate Equation 4 efficiently.

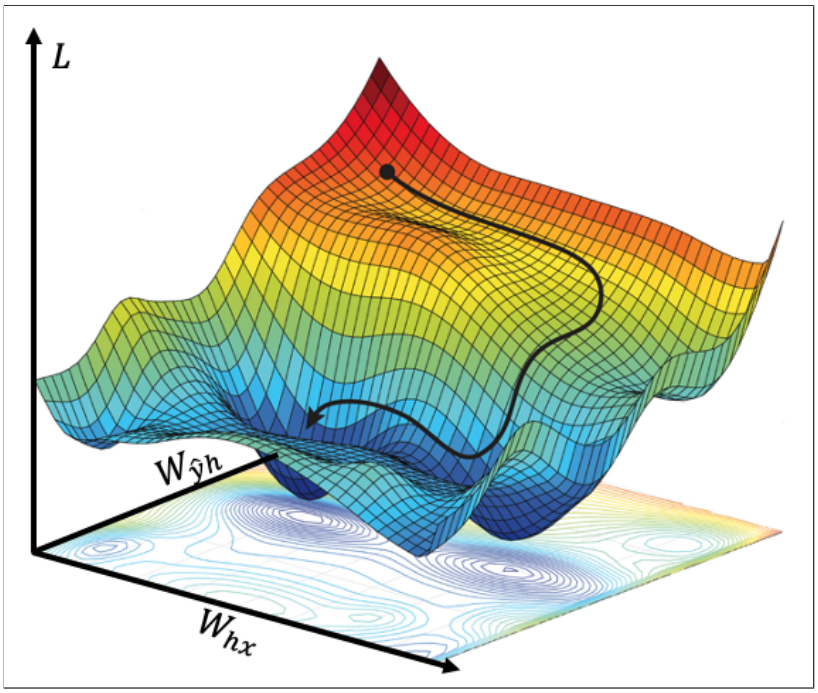

Figure 3: Optimization procedure of the parameter space often called *gradient descent* algorithm. Graphics after Hutson (2018)).

Once the direction of the steepest descent for all parameters θ is calculated we can adjust them accordingly. The update algorithm can be described as

$$\theta = \theta + \alpha \cdot \frac{dL}{d\theta}, \tag{5}$$

where α is often referred to as the *learning rate*. Figure 3 shows a visualization of this procedure for the 2-dimensional real case optimizing with respect to the weight matrices only.

2.2 Recurrent Neural Networks

Standard FFNN inherently yield a significant limitation. Input and output of the function are of fixed length allowing no variation. To emphasize this, take the task of machine translation. A sentence of an input language shall be translated to its corresponding counterpart in a target language. The input sentence could be in English, for example, *He goes shopping today.* Its French counterpart would be *Il fait ses courses aujourd'hui.* It becomes apparent very quickly that words between languages are not always perfectly aligned. Besides, the input lengths could be of different size every time.

To overcome this, we introduce a neural network architecture which processes inputs step by step (by the means of *time*). The output of this network will be produced step wise as well.

Definition A recurrent neural network (RNN) is a function $f : (\mathbb{R}^n)^+ \to (\mathbb{R}^m)^+$ which maps a variable sized input x to a variable sized output $\hat{y}$. The $+$ denotes this *variability* in a step wise manner (Wurm, 2018).

Forward Propagation Quite similar to the FFNN with one hidden layer, the input $x_t \in \mathbb{R}^{n\times 1}$ is multiplied by a weight matrix $W_{hx} \in \mathbb{R}^{k\times n}$ and projected onto a hidden layer $h_t \in \mathbb{R}^{k\times 1}$. Unlike a FFNN. this hidden layer has a second parameter. It is dependent on its previous state. Another weight matrix $W_{hh} \in \mathbb{R}^{k\times k}$ multiplied by the previous hidden state h_{t-1} is used to augment the current hidden state. Again, this hidden state is passed through an activation function a and multiplied by a weight matrix $W_{\hat{y}h} \in \mathbb{R}^{m\times k}$. For this timestep, this produces the output $z \in \mathbb{R}^{m\times 1}$. Formally, this can be expressed as

$$\begin{aligned} \hat{y} &= W_{\hat{y}h} h_t \\ \text{where } h_t &= a(W_{hx} x_t + W_{hh} h_{t-1}). \end{aligned} \tag{6}$$

It is important to note that all three weight matrices are shared amongst all time steps. For simplicity, biases are omitted this time. Figure 4 shows the enrolled computational graph of the RNN.

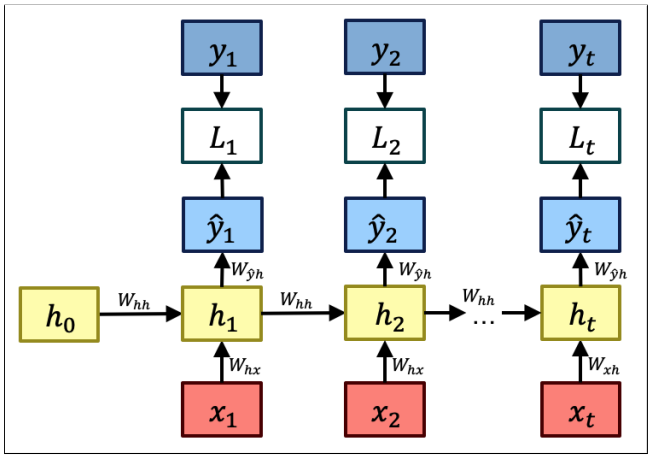

Figure 4: The computational graph of the RNN. h are the hidden states, x the inputs, $\hat{y}$ the prediction, L the loss function and y the ground truth.

Backpropagation Optimizing an RNN and computing its gradient is fairly simple and follows the procedure shown before. Using the computational graph we simply start at the rightmost hidden state and work our way backwards. The gradient for $W_{\hat{y}h}$ comprises no dependency in time (as can be seen in the figure). It follows that

$$\begin{aligned} \nabla_{W_{\hat{y}h}} L &= \frac{1}{t} \sum_{i=0}^{t} \left(\nabla_{W_{\hat{y}h}} L_{t-i} \right) \\ &= \frac{1}{t} \sum_{i=0}^{t} \left(\frac{dL_{t-i}}{d\hat{y}_{t-i}} \cdot \frac{d\hat{y}_{t-i}}{dW_{\hat{y}h}} \right) \end{aligned} \tag{7}$$

with an arbitrary objective function L.

Since we have no dependency in time, this is computationally inexpensive. Unfortunately, other gradient computations are not. We omit the derivation of W_{hx}'s gradient but show how one could obtain the gradient of W_{hh} (which becomes computationally intractable). Note, for the sake of simplicity we have omitted the activation functions for backward steps. Formally, the gradient of L with respect to W_{hh} is

$$\begin{aligned} \nabla_{W_{hh}} L &= \frac{1}{t} \sum_{i=0}^{t} \left(\nabla_{W_{hh}} L_{t-i} \right) \\ &= \frac{1}{t} \sum_{i=0}^{t} \left(\frac{dL_{t-i}}{d\hat{y}_{t-i}} \cdot \frac{d\hat{y}_{t-i}}{dW_{\hat{y}h}} \cdot \frac{dW_{\hat{y}h}}{dh_t} \cdot \frac{dh_t}{dW_{hh}} \cdot \frac{dW_{hh}}{dh_{t-1}} \cdot \frac{dh_{t-1}}{dW_{hh}} \cdot \frac{dW_{hh}}{dh_{t-2}} \cdot \dots \right) \\ &= \frac{1}{t} \sum_{i=0}^{t} \left(\frac{dL_{t-i}}{d\hat{y}_{t-i}} \cdot \frac{d\hat{y}_{t-i}}{dW_{\hat{y}h}} \cdot \frac{dW_{\hat{y}h}}{dh_t} \cdot \prod_{j=0}^{t} \left[\frac{dh_{t-i}}{dW_{hh}} \cdot \frac{dW_{hh}}{dh_{t-i-1}} \right] \right) \end{aligned} \tag{8}$$

For every time step t we have to go back t steps resulting in a computational complexity of $\mathcal{O}(t^2)$. Another issue is that while propagating through time gradients become increasingly small through shrinking eigenvalues, resulting in the *vanishing gradient problem* (Schmidhuber, Bengio, & Frasconi, 2003).

To tackle the above issues, various extensions to RNNs have been proposed, e.g. long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent units (Chung, Gulcehre, Cho, & Bengio, 2014). A further discussion, however, is beyond the scope of this section.

2.3 Encoder-Decoder Models

Again, the main idea of encoder-decoder models is influenced by human cognition (Schramm, 1954). When we communicate with each other one person thinks (encodes) of a situation and expresses (represents through a channel) this in linguistic terms. Another person listens (decodes) to this and can make sense of what the other person said. Figure 5 illustrates this procedure.

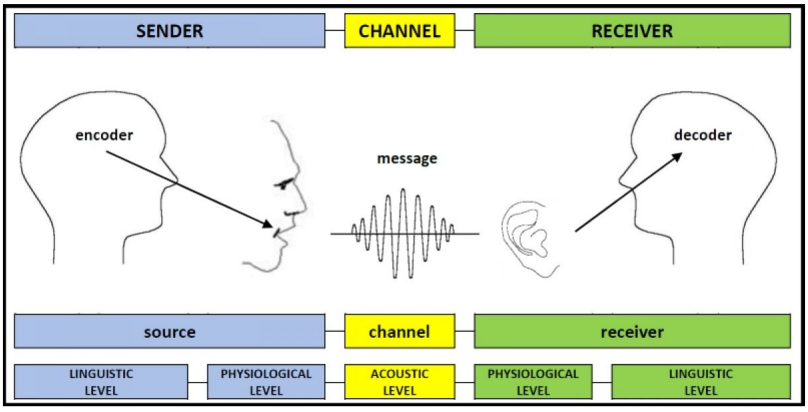

Figure 5: The Encoder-Decoder Model of Communication after Schramm (1954). Illustration after Williamson (n.d.).

Encoder-decoder models in mathematics (often referred to as *seq2seq models*), give a simple frame to sequential modelling approaches like RRNs or LSTMs. An encoder receives a sequence of an input language. This input is sequentially encoded, e.g., through an RNN. The RNN presents the final vector representation of the entire input sequence. In the previous example this would correspond to the linguistic expression. Only the very last emission $\hat{y}_t$ of the encoder will be further processed. The decoder (the listener) now uses $\hat{y}_t$ to gradually output a corresponding sequence in the target language. The sequence has been translated successfully.

Note that *language* does not necessarily have to correspond to linguistic languages. These types of models could, e.g., solve the task of named entity recognition. The input language would be an arbitrary human language and

the target language would be a annotated sequences of tags. Figure 6 shows the inner workings of encoder-decoder models.

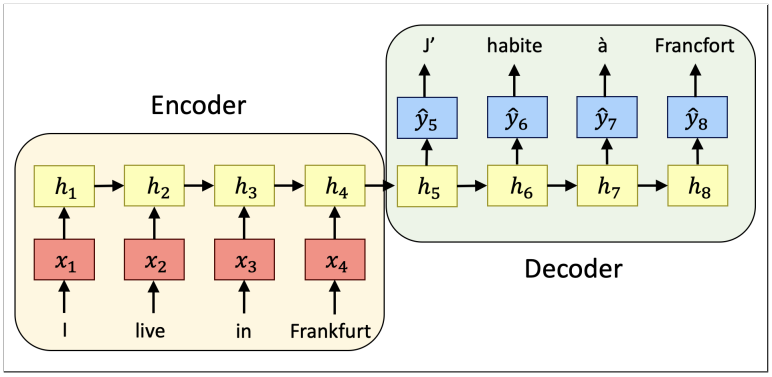

Figure 6: The Encoder-Decoder architecture applied to the task of neural machine translation (NMT).

In theory, these architectures encode sequences quite well, but in practice long-term dependencies are not captured well. This is mostly caused by the vanishing gradient problem (Schmidhuber et al., 2003). Besides, they are extremely non-transparent and an understanding for their fundamental functionality is simply missing.

2.4 Attention

Attention has been claimed to be an important concept of human stimulus processing by Psychologists and Neuroscientists for a really long time (Tang, Srivastava, & Salakhutdinov, 2013). As most technical advances, again, this idea has been borrowed from nature by the deep learning community.

First steps in implementing attention in deep learning models were explored by the computer vision community (Ba, Mnih, & Kavukcuoglu, 2014; Mnih, Heess, Graves, & Kavukcuoglu, 2014; Tang et al., 2013; Xu et al., 2015). Shortly after, the NLP community applied attention to, then, state-of-the-art encode-decoder architectures (Bahdanau, Cho, & Bengio, 2014; Kim, Denton, Hoang, & Rush, 2017; Luong, Pham, & Manning, 2015; Parikh, Täckström, Das, & Uszkoreit, 2016).

The addition of attention to encoder-decoder models resulted in significant performance boosts. Previous architectures encoded long sequences into only one vector representation, hence long-term dependencies were really hard to decode. Attention will remember the important part when encoding and decoding sequences, no matter where corresponding parts might be.

More importantly, attention enables deep insights into the workings of, to date, very opaque functionalities of neural networks. To an extent, one is finally able to explain which parts of an encoded sequence is paid attention to

when decoding a certain part of the output sequence. An example of this is nicely shown by utilizing neural machine translation (NMT). Bahdanau et al. (2014), for example, proposed one of the first attention mechanisms applied to encoder-decoder architectures. On one hand, they attained significant performance boost, specially for long sequences. On the other hand, they were able to align corresponding words of the input and target sequence. In essence, attention can be understood like a memory extension to the model. At every step the memory remembers the corresponding parts of importance. Illustrations of the attention matrix as the workhorse of this process can be seen in Figure 7.

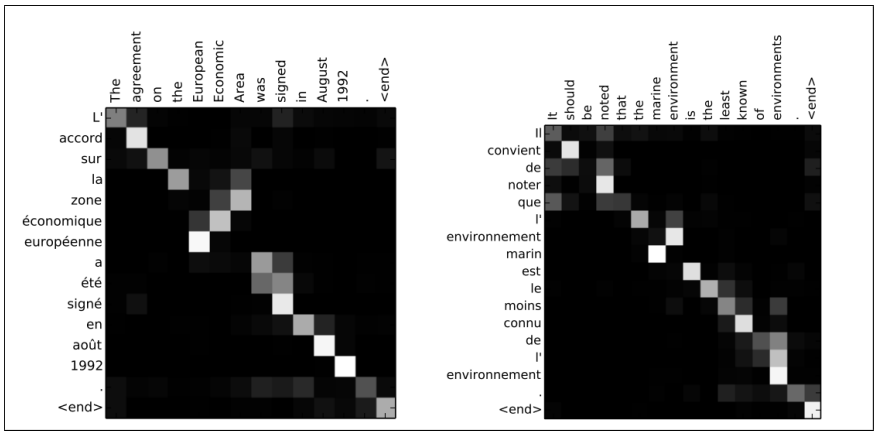

Figure 7: Alignment of words applied to the task of NMT. One can see to which parts of the target sequence attention is paid to in the corresponding input sequence. Illustration after Bahdanau et al. (2014).

2.5 Transformers

While attention was, without a doubt, an important new discovery, the heavy burden of encoder-decoder architectures was still inseparably associated with it. Besides the issues already discussed, they yield another limitation. Since these models are of sequential nature they entail non-parallelizability. Though workarounds have been proposed (Kuchaiev & Ginsburg, 2017), the core of the problem stays the same. A paradigm shift from encoder-decoder architectures to so-called transformers has been enforced by Vaswani et al. (2017)'s pioneering paper *'Attention is all you need!'*. The following subsections present the main ideas of the paper. Annotations complementing some of the concepts which have come off too shortly, are added in addition.

Model architecture The architecture inhabits one critical advantage over previous architectures. That is, sequences do not enter the model sequentially. Inputs are encoded at once. In 8 one can investigate the high-level architecture of the model. The model consists of an encoder (left side) and a decoder

(right side), where both consist of N stacked attention layers followed by simple FFNNs. Each layer is expanded by residual connections enabling different abstraction levels of information (He, Zhang, Ren, & Sun, 2015). Every output is concatenated with the unprocessed information and normalized. The encoder then passes the encoded input sequence to the decoder. The decoding process is identical to the encoding process, except the decoder uses masked multi-head attention in the first step. In principle this means that the right part of the sequence, which has not yet been processed, is disguised. This is to retain the auto-regressive (left-to-right) behaviour of the model and prevents peaking to the right side. Finally, the decoded sequence is fed into a linear layer and softmaxed to attain a probability distribution over a possible output sequence.

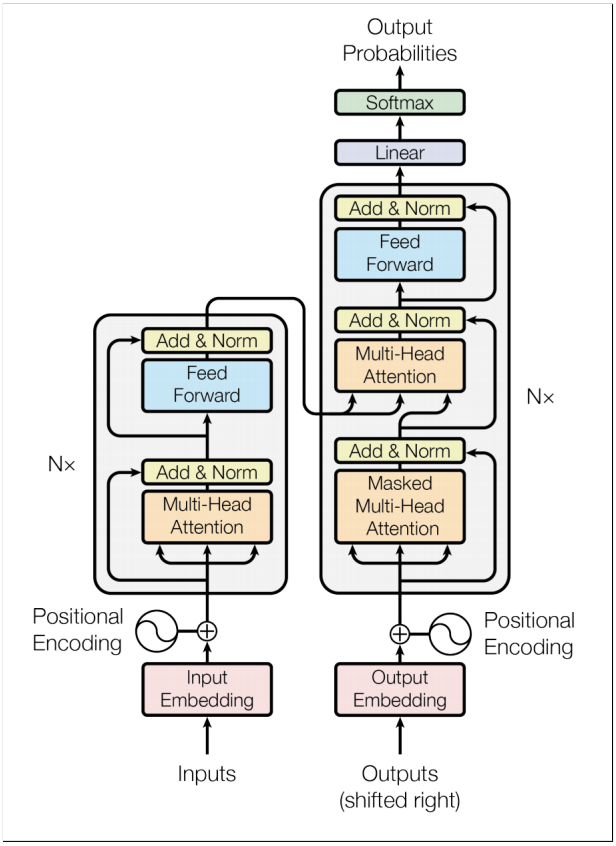

Figure 8: Transformers model after Vaswani et al. (2017).

Scaled Dot-Product Attention In essence, attention in the transformers model can be seen as a selection mechanism retrieving the most important information of an input sequence.

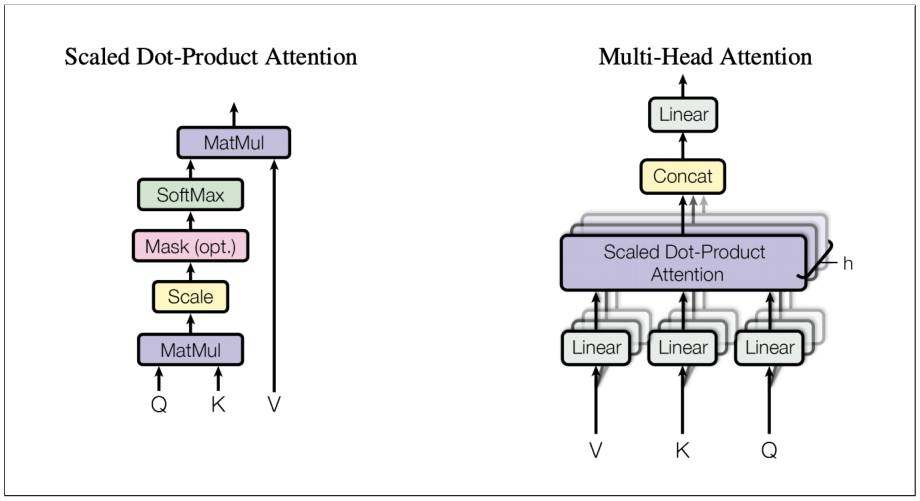

Figure 9: Attention architecture after Vaswani et al. (2017).

Matrices in the following should be seen as batched inputs of their respective subject. A query matrix Q is multiplied by a key matrix K. Vectors (the rows of Q and columns of K) which are close to each other result in a large value. This is followed by a scaling and softmax. This procedure yields a sparse matrix which can be seen as a selector to the value matrix V. It is important to note that Q, K and V are the result of the input matrix X and the respective weight matrices W^Q, W^K and W^V which are the subject of the training process (e.g., $W^Q X = Q$). More formally the scaled dot-product attention can be expressed as

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d}})V \tag{9}$$

where d denotes the dimension of the input embeddings.

In short, we use the Query Q to select the key K. This key K is used to select the appropriate value one should pay attention to.

Multi-Head Attention In a simplified manner, multi-head attention can be understood as the concatenation over h scaled dot-product attention modules, referred to as *head*. Besides shared weights across all heads, each $head_i$ is initialized with their own weight matrices W_i^Q, W_i^K, W_i^V resulting in a mathematical expressing as

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W^O \tag{10}$$

$$\text{where } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{11}$$

$$= Attention(XW^Q W_i^Q, XW^K W_i^K, XW^V W_i^V). \tag{12}$$

The transformers model include a few more important aspects. A more thorough explanation (e.g., discussing the positional encodings) is beyond the scope of this thesis and is therefore omitted.

3 Fundamentals of Information Retrieval

In the early days only librarians, intellectuals and professional researchers would be in the need of specific information aimed at filling a knowledge gap. Since the emergence of the web, most of us are confronted with the concept of information retrieval (IR) every day. Introducing the task of IR gives rise to taking a fine-grained look on the words compiling the concept, namely *information* and *retrieval*.

Information. Information can be understood as knowledge for targeted preparation and execution of actions in context (Kuhlen, 1990, 2013).

Retrieval. The concept of retrieval can be understood as the process of finding something (Cambridge Dictionary, 1999b).

We can conclude that IR therefore can be seen as the process of finding knowledge for targeted execution of an action. By this definition, looking for a phone number inside a phone book, because we want to call a friend, would be considered as IR already. As the previous definitions are very general, they give reason to be expanded to a more specific use case. In the context of computer-based methods, information retrieval can be defined as finding information in a large corpus of unstructured documents that satisfy some information need (Manning, Raghavan, & Schütze, 2008). Unstructured data often refers to textual data. While for humans documents and sentences may be logical and coherent, a computer is faced with the difficulties of syntax, semantics, pragmatics, even common-sense and contextual knowledge. For humans, this is fairly simple as we learn all of the former from being a child. The latter is learned during our day-to-day existence. For a computer it is a very hard task as formalizing the entirety of the prior remains unsolved until today.

IR in practice has been subject to scientific research ever since the 1940s and 50s. While not using the term IR in specific, the idea of the collective mind and its ability to remember, obtain and understand critical information with high speed and and flexibility may have been the birth of modern IR (Bush, 1945). Shortly after, though describing tape-based approaches, the idea of searching for information on a computer was proposed for the very first time (Holmstrom, 1948). Modern information retrieval was eventually introduced by Gerard Salton's research group at Cornell University in the 1960s. Until today, the principles of modern information retrieval rely on their fundamental work (Mikolov, Chen, Corrado, & Dean, 2013).

3.1 Conceptual Model

While practical information retrieval knows many different approaches, a conceptual model of information retrieval can be constructed in a relatively simple manner. This can be seen in Figure 10.

Induced by a knowledge gap a user formulates their information need into an explicit query. An arbitrary retrieval function of the query and the document representations returns potentially informative material to the user. The information returned may (or may not) influence the user's information need and they may (or may not) formulate a new query based on the acquired knowledge.

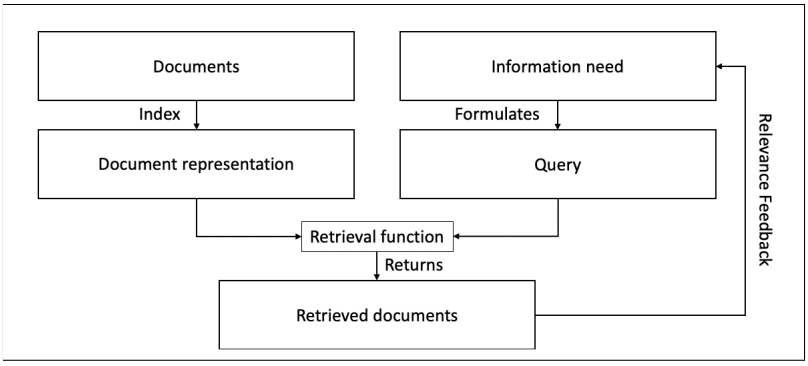

Figure 10: A conceptual information retrieval model after Lalmas (2011).

While Figure 10 provides a good starting point for a general framework it may be too general and leaves out important parts of the equation utilized by this work. Of course, queries do not need to be indexed, but with a view to today's IR approaches, formatting the query to the needs of the the retrieval function is certainly needed. More importantly, however, is the information loss when explicitly formulating the query. A discrepancy between the user's real intent and the explicit query is governed by **1.** The ambiguity of language and words in general, **2.** choosing (possibly correct) words which may not occur in target document(s) and **3.** imprecise formulation (and possibly incorrect words) caused by missing background knowledge.

As an extension to the conceptual model shown in Figure 10 a new conceptual framework is provided in Figure 11. This framework forms the rationale for the approach utilized in Section 5.

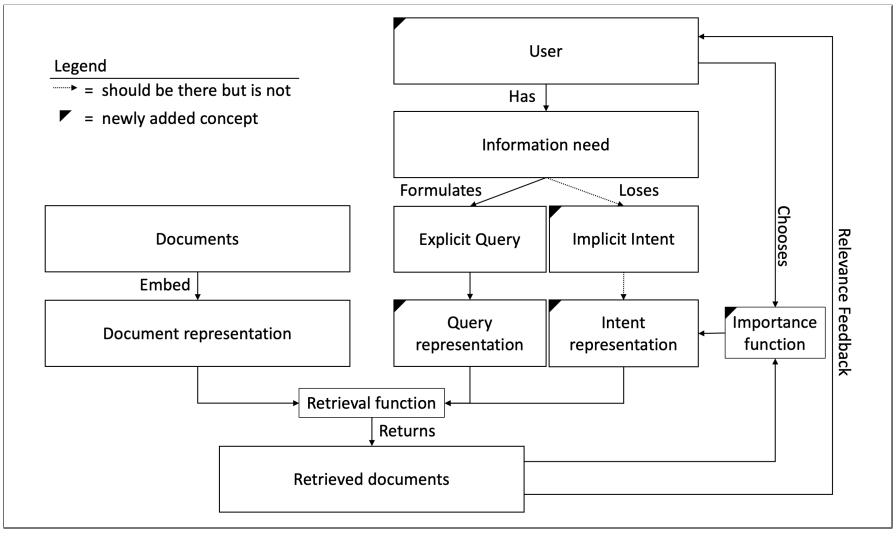

Figure 11: An extended conceptual information retrieval model.

The user is added to the view since they are the center of all actions. They user is the recipient of the retrieved documents and their potential information value. The user explicitly chooses documents relevant to their need (which is important for this work). Therefore, the importance function of the choice and the retrieved documents is added to the model. This function produces an intent representation which *might* have been the user's implicit intent (if we were able to capture it). The retrieval function now is not only a function of the document and query representation, but also a function of the intent representation. This intent will be formalized by the means of *context* in Section 4 and eventually used in Section 5.

3.2 Information Retrieval Models

IR approaches in practice are manifold. We classify different models into three distinct classes, based on their mathematical properties. These classes are of set-theoretic, probabilistic and algebraic nature (Kuropka, 2004). Note that borders between the classes are blurring. Approaches may combine various aspects. It is therefore not always possible to clearly assign specific approaches to only one class. The field of IR to this day is very broad. This section shall not give the reader a deep understanding of IR models, but rather shed light on the methods in a horizontal manner. We briefly summarize the key aspect of the respective models and point to relevant literature. A thorough (vertical) explanation of the model type used in this work, namely a neural algebraic approach, is described in depth in Sections 3.3 - 3.7.

The class of **Set Theoretic Models** comprises

1. Boolean Models, which simply vectorize documents' term counts and query the data in a boolean exact-match manner (Hiemstra, 2009).

2. Extended Boolean Models, which counteract unranked results, possible large sizes of outputs and counter-intuitive results of the above (Lee, 1994; Salton, Fox, & Wu, 1983).

3. Fuzzy Models, which assume fuzziness of objects' properties and introduce the concept of non-binary and continuous characteristics (Crestani & Pasi, 2000; Kraft, Bordogna, & Pasi, 1994).

The second class of importance are **Probabilistic and Statistical Models** which can be summed up as

1. Language Models, which try to capture the probability of a word occurring in a context (Manning et al., 2008). An example of this approach (mixed with algebraic properties) can be seen in Section 3.6.

2. Binary Independence Models, which uses the simple assumption that documents are represented as binary vectors of term occurrence. By the means of bayes' rule relevance rankings can be conducted in a probabilistic manner (Manning et al., 2008). An example of this approach (mixed with algebraic properties) can be seen in Section 3.4.

3. Other Graphical Approaches (Manning et al., 2008).

3.3 Neural Information Retrieval

With advances in deep neural networks architectures for natural language processing (NLP) (Devlin et al., 2018; Peters, Ammar, Bhagavatula, & Power, 2017; Vaswani et al., 2017), the availability of large data sets (e.g., arXiv Bulk Data, DBpedia, SQuAD) and a new generation of hardware longstanding approaches like BM25 get serious competitors (Robertson & Zaragoza, 2009). Neural IR can be broadly classified into two classes (Mitra & Craswell, 2018). Either representations of the query and the documents are learned in a supervised or unsupervised fashion (which this work will make use of), or ranking itself is learned (Hui, Yates, Berberich, & de Melo, 2017; Liu, 2009). Both approaches have been explored and tested thoroughly. Ranking models of the later usually learn matching by being discriminatively trained on large amounts of query-document pairs (Liu, 2009). Neural methods based on feature representations focus on constructing vectors which capture latent semantic knowledge of the sentences encoded. These vectors are often referred to as *embeddings*. As this thesis makes use of these embeddings a thorough explanation of this approach is given in Section 3.5.

3.4 Vector Space Models

Vector space models (VSM) are algebraic models which represent words and text documents in the form of vectors in an n-dimensional vector space. The underlying idea of this approach is that simple algebraic rules can be conducted to retrieve relational semantic information between representations (Salton, Wong, & Yang, 1975).

Consider a corpus (a collection) C of documents with n distinct words over all documents. Then each document will be represented as an n-dimensional vector over this corpus. The j^{th} value of the vector denotes the importance of the respective word. In the original approach proposed by Salton et al. (1975), each value of the n-dimensional vector representation is the product of a local (the term frequency, tf) and a global (the inverse document frequency, idf) variable, resulting in a statistical measure called *term frequency-inverse document frequency*, short *tf-idf*. In mathematical terms, we can express a document representation d of document i as a vector of this statistical measure as

$$d_i = [\, tf(w_1, d_i) \cdot idf(w_1, C),\ \ldots\ ,\ tf(w_1, d_i) \cdot idf(w_n, C)) \tag{13}$$

where $tf(w_j, d_i)$ describes the raw count of the word j in document i. $idf(w_j, C)$ denotes the importance of the word j relational to the corpus C. More precisely

$$idf(w, C) = log\left[\frac{|C|}{|\{d \in C \mid w \in d\}|}\right] \tag{14}$$

where $|C|$ denotes the cardinality of the corpus and the denominator can be read as the amount of documents d in the corpus C which inhabit the word w. Note that idf is an important measure how much information certain words provide. If a word occurs very often across documents (likely a word like 'the' or 'and') the denominator grows and idf's value becomes small. Therefore, we should not pay much attention to that word, resulting in a small value for *tf-idf*.

Classic vector space models have been widely adopted since they were first introduced by Gerard Salton's IR group. They form the foundations of today's vector models (Manning et al., 2008; Mikolov, Chen, et al., 2013; Mitra & Craswell, 2018). The elementary idea of these models imply several very important advantages. Simple linear algebraic rules can be conducted to make sense of the representations and their relations (Coecke, Sadrzadeh, & Clark, 2010). This allows conducting different, though all proportional, similarity measures to get a sense of how terms, queries and documents relate to each other. Unlike boolean relevance ranking we can finally construct a non-binary relevance ranking which enables partial matching.

VSMs in the form presented are very powerful, but they imply significant limitations which have been the subject of subsequent research. While the weighting is quite intuitive and can be explained in a simple manner, important properties of language are ignored.

1. **Invariance of Language:** Semantics may not be captured due to the invariance of language. It is possible to express the exact same thing using

completely different words (Winston, 2014), e.g., *The beetle crawls* versus *A bug scuttles*. A representation of the former in a VSM results in *no similarity at all*, which obviously is incorrect.

2. **Word Order:** The order of words is completely ignored assuming independence amongst them. This has been one of the hardest tasks to counter and only recently promising approaches have been proposed (Dai et al., 2019; Devlin et al., 2018; Ke, He, & Liu, 2020).

3. **Document length:** Though not related to language properties, the document length limits the accuracy of the approach. Consider a sequence of word *The beetle is an insect* and another sequence of n words talking about beetles. It is easy to show that with $n \to \infty$ the similarity of the documents converge to 0, which obviously is incorrect.

In the course of this analysis it becomes clear that the main challenge for VSMs is finding an appropriate language model which represent the complexities of semantics in natural languages. Starting with the pioneering work of Bengio, Ducharme, Vincent, and Janvin (2003) many impressive approaches boosting research in the field of word embeddings have been proposed (Mikolov, Chen, et al., 2013; Mikolov, Sutskever, Chen, Corrado, & Dean, 2013; Pennington, Socher, & Manning, 2014).

3.5 Word Embeddings

In essence, word embeddings are a mapping from a sequence of words to an n-dimensional vector of real numbers. The vector representations are supposed to capture the meaning of a word such that vectors which are close to each other comprise related, if not the same, concepts. The approach shown in the previous section only embeds sequences of words. Inspired by this, one could one-hot encode single entities. Unfortunately, this yields several limitations. The vectors produced would be as big as the vocabulary's size resulting in a huge and sparse space. Besides, every two vectors would be orthogonal to each other. This would result in a similarity of 0 for all vector pairs.

Other approaches try expanding contexts of a word by utilizing hierarchical linguistic resources such as WordNet (Gong, Cheang, & Hou, 2005). Hypernyms (generalized term), hyponyms (specific terms) and synonyms (same terms) are explored to overcome invariances in language (Ono, Miwa, & Sasaki, 2015). While the approaches certainly make sense and expand the concepts of a single word, similarity still cannot be retrieved.

To better capture semantics in word representations, novel deep neural network (DNN) architectures have been proposed. These DNN approaches learn textual representations in a supervised or unsupervised manner. Unlike the *tf-idf* approach, the embeddings produced by these approaches usually represent vectors in latent, low-dimensional, continuous, and dense space. Here, latent refers to a space which is hidden or unobservable. The axes of these vectors do not represent specific words, therefore one can impossibly make sense of these

representations. Most of today's approaches produce vectors of sizes between 25 and 1000, which is much less then the underlying vocabulary size of a given corpus, hence the low-dimensionality (even though from a human-imageability perspective these vectors are high-dimensional). The values of the vector are realized as a real non-zero value resulting in dense representations.

A variety of neural word embedding approaches have been explored and deployed in the recent years. Based on the foundations of the pioneering work of (Bengio et al., 2003), the most adopted approaches used in practice are **1.** the Continuous Bag-of-Words and Continuous Skip-Gram Model (CSG) (Mikolov, Chen, et al., 2013; Mikolov, Sutskever, et al., 2013), **2.** GloVe (Pennington et al., 2014) **3.** ELMo (Peters et al., 2018) and most recently with the rise of Attention in NLP **4.** BERT vector representations (Devlin et al., 2018).

In the following, the CSG models are explained in depth because they set a milestone for today's most successful approaches. Besides it is supposed to build an intuition (and the mathematics) around building effective word vector representations.

3.6 Continuous Skip-Gram Model

The Continuous Bag-of-Words Model tries to optimize a target word given the context, whereas the the Continuous Skip-Gram (CSG) Model's objective is to maximize the probability of a context given a target word (Mikolov, Sutskever, et al., 2013). Figure 12 provides an idea of this optimization objective. For an introduction to deep learning please refer to Section 2.

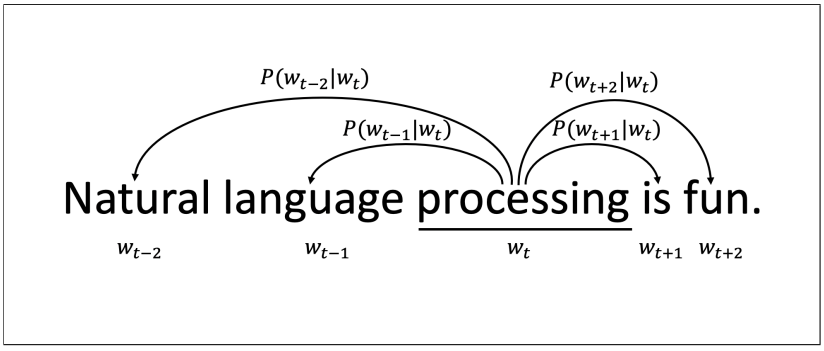

Figure 12: The learning objective of the CBOW model.

Every word in a given vocabulary is encoded as an n-dimensional real vector twice. One encoding describes words as target words and one encoding describes words as context words. This may seem unusual, however, encoding twice yields nice mathematical properties as we will see in the following.

Forward propagation is quite straightforward and consists of a two-layer neural network followed by a $softmax$. The input of the network is given by the one-hot encoded index vector $w_t \in \mathbb{R}^{n \times 1}$ of the current target word t. Multi-

plying w_t with the word embedding matrix $W_{emb} \in \mathbb{R}^{d \times n}$ acts like selecting the correct representation of t and yields the word embedding $v_t \in \mathbb{R}^{d \times 1}$. Multiplying v_t with the context matrix $W_{con} \in \mathbb{R}^{n \times d}$ yields $y_{t+j} \in \mathbb{R}^{n \times 1}$ for the j^{th} word surrounding the target word. y_{t+j} is now turned into a probability distribution over the vocabulary using the $softmax$ function yielding $P(w_{t+j}|w_t)$. This probability distribution is now compared against the true index representation. Optimizing the discrepancy between prediction and ground truth can be achieved by e.g. minimizing the negative log likelihood w.r.t. (both!) word matrices W_{emb} and W_{con}. More formally, forward propagation can be expressed as

$$P(w_{t+j}|w_{w_t}) = \sigma\left(y_t\right) \tag{15}$$

$$y_t = W_{con} \cdot v_t \tag{16}$$

$$v_t = W_{emb} w_t \tag{17}$$

where the $softmax$ function σ is given by

$$\sigma(x_i) = \frac{exp\left(x_i\right)}{\sum_{j=1}^{n} exp\left(x_j\right)} \tag{18}$$

and objective function to minimize could be expressed as the negative log-likelihood, that is

$$L(W_{emb}, W_{con}) = -\frac{1}{n} \sum_{t=1}^{n} \sum_{\substack{-m \leq j \\ j \leq m \\ j \neq 0}} log\ P(w_{t+j}|w_t) \tag{19}$$

with m expressing the radius of context words to predict.

Note that the computational complexity of the objective function is given by $\mathcal{O}(m \times d \times log(n))$, where m is the radius, d is the embedding dimension and n the vocabulary size. The computation of the gradients becomes increasingly more intractable with growing vocabulary size. Of course one does not want to accept loss of quality due to low dimensional vectors or a small radius. In a subsequent paper (Mikolov, Chen, et al., 2013) therefore introduced two highly scalable and efficient approximations which boosted research in word embeddings significantly. An explanation of these, however, would be beyond the scope of this section and is omitted.

3.7 BERT

Now that we have discussed one of the early breakthroughs of word embeddings we take a big leap and continue with the presentation of the word embeddings used in this work.

BERT, short for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers, is a novel multi-purpose language model architecture which is able to outperform recent systems significantly (Devlin et al., 2018). BERT also attests a new

development amongst novel NLP models, namely designing systems which can be applied to multifaceted range of tasks with relatively low effort (Brown et al., 2020; Yang et al., 2019). The original approach by itself resulted in new state-of-the-art results in 11 NLP tasks.

Architecture The model layout consists of two, very similar, steps. Step one, the pre-training, consists of feeding unlabeled tuples of subsequent sentences to the model. The objective BERT tries to solve during pre-training time is composed of two tasks. Either the masked words of the sentences are predicted, or, given sentence 1, sentence 2 is predicted. Step two, the fine-tuning, initializes the model with the parameters from pre-training and, once again, gets fed tuples of sequences. This time, however, the tuples try to solve a very specific task. For instance, BERT could be fine-tuned to the NLP task of question answering. The input would then be a set of *(question, context, answer)* tuples.

Previous models were only trained in an auto-regressive manner (either right-to-left or left-to-right), being able to only see one side of the context. BERT improves this with its fully connectedness by letting the model see the whole context the entire time. This, however, implies an important limitation. Since the model is informed about the whole context the entire time, *predicting* the next word of a sequence would result in *knowing* the next word of the sequence. No matter at which time step it is, the model would, in simple terms, see itself. To counteract this, during training time (and later during run time) a certain percentage of the tokens are masked. E.g., the sequence *'The cat likes to play'* could result in *'The cat [MASK1] to play'*. Besides this, there are a few other artificial tokens added to sequences, but a further explanation will be omitted as they are not too important to grasp the concept of the model.

The architecture of BERT is very simple and its name already reveals what lays behind it. It is comprised of a fully connected L-layer bidirectional transformer encoder neural network with A attention heads and a hidden size of H. 3.7 shows a high-level illustration of pre-training and fine-tuning the model. For an explanation of transformers refer to 2.5.

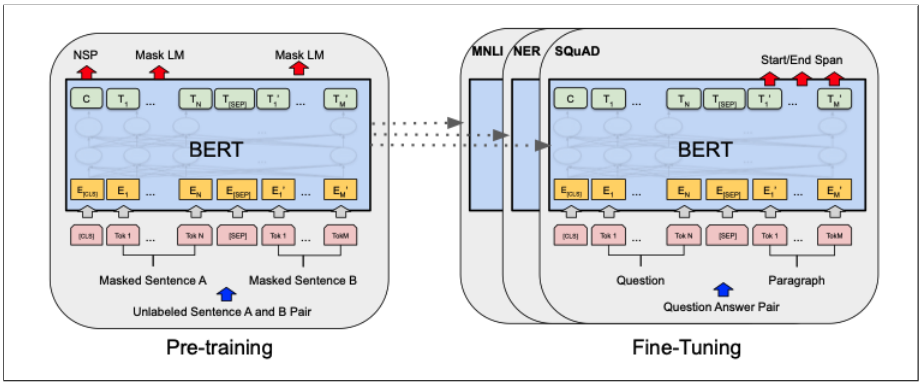

Figure 13: The two steps of training the BERT model after Devlin et al. (2018).

3.8 Sentence BERT

Retrieving word embeddings from BERT, and much more importantly sentence embeddings, has yet to be discussed. The procedure is quite simple. For each word embedding, the mean of the (often four) last layers of the model are taken. This was proven to be an effective way to retrieve semantically meaningful word representations (Ethayarajh, 2019).

To create sentence embeddings of arbitrary length, Sentence-BERT (SBERT) is being used. SBERT simply adds a pooling operation to the output of the BERT model. This results in a mean representation of all words in a sentence. Semantically meaningful representations are computed and can be geometrically compared. Reimers and Gurevych (2019) introduce SBERT and provide an effective and efficient API this work makes use of.

4 Context in Information Retrieval

Context in its general form gives cause for discussion. One can describe it as the the conditional interrelations of an event to its surrounding (Merriam-Webster, n.d.). Another description might be, the surrounding explanation for the existence and the being of an object (Cambridge Dictionary, 1999a). In other words, by these definition the true appearance of an object is conditioned on the environment it lives in. Mylonas, Vallet, Castells, Fernández, and Avrithis (2008) frame the prior as the relative nature of truth.

Disciplines strongly connected to artificial intelligence make use of capturing and understanding context. Cognitive modelling (Nirenburg et al., 2018; Preuveneers et al., 2004), information retrieval (IR) (Ruthven, 2011; Shen, Tan, & Zhai, 2005) or recommender systems (Adomavicius & Tuzhilin, 2011) can be significantly enhanced by understanding the user's context. For applications like the internet of things, its importance has been recognized as well(Perera, Zaslavsky, Christen, & Georgakopoulos, 2013).

The concept of context is most easily explained with an example of natural language in the context of IR. Assume a person employs an IR system (e.g., Google) to fill their knowledge gap. They enter the word *'mercury'* and without further investigation it is unclear what they want to find out. The word *'mercury'* is ambiguous in many different ways and leaves room for interpretation. A chemist is most likely interested in the chemical element, an astronomer wants to know more about the planet and a musician is interested in their idol. Without explicitly naming *'planet'*, *'element'* or 'singer' an extension of the query would already give cause for a more precise understanding. If one were to type *'reaction mercury aluminum'* the context of *'mercury'* becomes clear. It becomes evident quickly that user queries may not provide enough information to return satisfactory search results.

Initial approaches to utilizing *context* have been based on statistical language models (Bharat, 2000; Finkelstein et al., 2001). In the last years, systems based on implicit feedback have moved into the spotlight. While this idea has its origins in recommender systems (Oard, Kim, et al., 1998) its increasingly finding popularity in IR systems (Shen et al., 2005; White, Jose, & Ruthven, 2006).

Closing a complex information gap is difficult. Many iterations are necessary to find the right information (Vallet, Fernández, Castells, Mylonas, & Avrithis, 2006). Entering a query, selecting few documents, discarding many, which documents are read and how are they read reveal important indications about the user's intent. This observation and adaption procedure is often referred to as relevance feedback (Drucker, Shahrary, & Gibbon, 2002). Using this knowledge in an on-line fashion, one can enhance the content personalized to the individual's needs. Enhancement takes place by augmenting the explicit user requests with the user's implicit preferences (Gauch, Chaffee, & Pretschner, 2003).

One fairly simple approach to describing context in information retrieval comprise *taxonomies of context*. They provide a clean framework to encapsulate the entirety of context and facilitate further analysis.

4.1 Taxonomy of Context

Taxonomy in general describes the classification of objects and concepts including their underlying interrelation. Each class in a taxonomy comprises inclusiveness to another (Atran, 1993). This implies taxonomies to be of hierarchical nature. Taxonomies of context in IR have been explored extensively by the scientific community as such (Myrhaug & Goker, 2003; Tamine-Lechani, Boughanem, & Daoud, 2010).

Myrhaug and Goker (2003), for example, make use of the user's *personal* and *social* context. They view the *task* and consider the *physical* and *geospatial* environment. These two are important as the system they proposed is supposed to interact with tourists and shall provide tailored content.

Ingwersen and Järvelin (2005), on the other hand, want to provide a generic frame to context in IR and point to possibly promising research directions. They assign inter- and intradocument properties an important role. Inter-document properties may refer to, e.g., the topic, paragraph structure or the generic story behind a paper (Winston, 2014). Intra-document properties may refer to references, citations, connections to other knowledge bases. They even define the user handling of the IR system as context.

The dimensions such taxonomies can adapt to are manifold. As tech merges with everyday life, and slowly becomes part of us, utilizing the potential of context has not yet reached its limit. By the two previous authors, it seems clear that it is common practice to design taxonomies of context adapted to the system's specific needs. We therefore discuss the basis to this system's approach.

Many of the ideas presented in the following are based on Boughareb and Farah (2014)'s excellent review on taxonomies of context. They provide a broad overview of previously proposed taxonomies and reference them for further analysis.

4.2 Taxonomy used in this work

The taxonomy used in this work forms the rationale of the context provided to IR system shown in Section 5. Note that the following discussion primarily evolves around the COVID-19 pandemic. Assumptions or statements are made to the best of our knowledge by the time of writing this thesis (September 2020). Figure 14 shows an illustration of the taxonomy discussed thoroughly in the following.

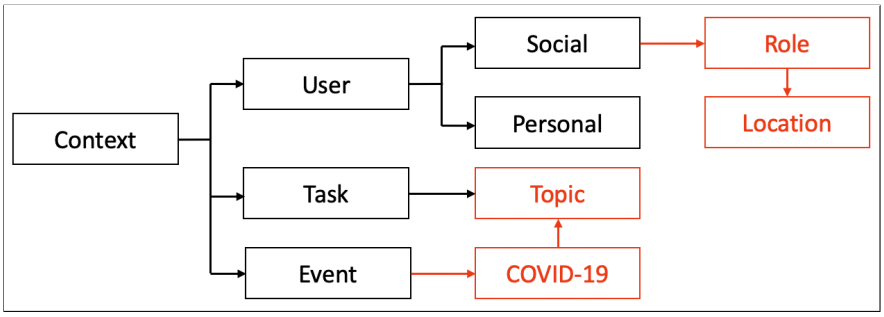

Figure 14: The taxonomy used in this work. Black nodes denote properties previously describes by other authors. Red nodes form the extension explained by this work and used throughout the thesis.

User Context: the Role User properties are always part of a context taxonomy (Boughareb & Farah, 2014). The dimensions in which they reveal themselves, however, differ. Some claim physiological and demographic aspects such as pulse, blood pressure, eye movement movement, age, gender to be of importance (Ingwersen & Järvelin, 2005; Myrhaug & Goker, 2003; Tamine-Lechani et al., 2010). It is notable that social aspects of the user find consideration in all current taxonomies of context (Boughareb & Farah, 2014). This suggests to pay special attention to this part of the user's context.

Indeed, in the context of the COVID-19 pandemic the social aspects of the user are critical factors informing the IR system. The following examples may seem trivial, but they show how a person's role implies possible locations and topics of interest. The also provide evidence for the dynamics of role-changing.

Consider a person who takes their children to school in the morning (role: parent). Next, the person aids their parents with grocery shopping (role: elderly carer). After lunch, the person does a workout at the nearby gym (role: athlete). In the evening, they go to their night shift at the airport and fly to Rome (role: pilot). A person's role changes dynamically with the blink of an eye. An IR system should be able to adapt to these dynamics quickly. In the long run the system should understand which roles a certain user has taken before such that it can switch between them accordingly.

User Context: the Location The location of interest is an important factor the IR system needs to grasp. The dynamics of the pandemic is highly sensitive to the location it is acting in (Gladwell, 2002). For example, the aerosol spread is of different nature in a warehouse with poor ventilation (or even beneficial air circulation resulting in a perfect spread) compared to an open air sports field. Safety precautions differ between schools, retirement homes or hospitals.

These locations are governed by, or conditioned on, the role the user takes the very moment asking. Assuming one knows the role of the user, a system should be able to narrow down possible locations of interest efficiently. E.g., assume the IR system was aware the person asking acts as a parent during the

search session. The system should be able to reason locations of interest like schools, playgrounds, sports fields, the own and other homes. If one asks as an *athlete*, they are likely interested in gyms, saunas or (outdoor) sports places. If one asks as a *pilot* they are likely interested in air planes or the waiting area before the flight. While this inference process may be of deeper importance, we were not able to formalize and investigate the problem due to time constraints. Location context in general, however, plays an important role in the experiments conducted in Section 5.

Task Context: the Objective The user's objective needs be taken into consideration when analyzing context (Ingwersen & Järvelin, 2005; Myrhaug & Goker, 2003; Tamine-Lechani et al., 2010). If the system knows what the user might be interested in upfront, possible search results can be narrowed down to their interest. (Myrhaug & Goker, 2003) state that the task can be explicit goals, actions or events. In the context of the COVID-19 pandemic, this objective can be framed as **1.** minimizing risks associated with the dynamics of the viral process and **2.** the direct causes of this process. To gain a deeper understanding of the prior two, we thoroughly analyze generic properties of pandemics and apply them to the special use case of the COVID-19 pandemic. If not stated otherwise, all information regarding pandemics are taking from Madhav et al. (2017).

Event Context: the Pandemic Though, the current situation is new to most people, pandemics and epidemics have been thoroughly studied since the early days of modern medicine (Barry, 2005). Pandemics yield different complexities on many different levels. In the following, the state of the COVID-19 pandemic is analyzed with respect to its viral category and the current time period. While these categorizations may not seem necessarily useful for the IR system in this work, they deliver an extendable frame for future scenarios. Plus, they form the rationale for deriving the two important categories, namely *mitigation measures* and *pandemic impacts*.

Viral Categorization According to Madhav et al. (2017) one needs to distinguish between the pathogens causing epidemics or pandemics. There are three classes of viruses which have different probabilities of causing global pandemics. A pathogen's lethality, morbidity and the spread mechanism are decisive properties for the categorization.

The first group includes influenza viruses, and at the very latest, certain strains of corona viruses. Their pandemic potential can be framed as **1.** efficient (aerosol) transmissibility, **2.** long contagious (asymptomatic) periods **3.** relatively mild symptoms (if any), **4.** low lethality and **5.** difficulty in diagnosis (due to overlaps with symptoms caused by other pathogens). Group two includes pathogens which inhabit the same properties as group one, however, have not developed human-to-human transmissibility yet. As viruses are known to mutate very quickly these types of pathogens yield a considerable high risk

of adapting and causing outbreaks. Group three, with relatively low chance of causing pandemics, consists of viruses which **1.** do not spread efficiently, or **2.** get detected quickly (e.g. causing unique diseases or killing quickly).

A viral distinction does not *have* to be conducted with respect to their chance to cause a global pandemic. It is, however, a reasonable approach as measures taken between pandemics caused by group one versus group three pathogens vary clearly.

Pandemic Periods Madhav et al. (2017) suggest to classify the periods of the pandemic chronologically into three different stages. The state of a pandemic can be characterized as following. **1.** The prepandemic period: the period of time before a pandemic starts. Countermeasures include, but are not limited to, creating situational awareness, monitoring human and animal diseases developments, resource stockpiling, extensive planning and preparation. **2.** The spark period: the period of time where a pandemic begins. Countermeasures include, but are not limited to, initial outbreak detection, pathogen classification, continued surveillance of human and livestock infections, extending communications management, rolling out severe measures such as quarantines, isolation and contact tracing. **3.** The spread period: the period of time where a pandemic is fully underway. A description follows below, as this is the time period of the COVID-19 pandemic by the time of writing this thesis.

Pandemic Impacts Pandemics of group one in time period 3 cause severe changes in all areas of life. In essence, one can frame these into health, economic, social, cultural, political impacts. As a knowledge base for a list of topics serves the Wikipedia article *Impact of the COVID-19 pandemic* (Wikipedia contributors, 2020b). As the list is very extensive, topics have been handpicked and can be observed in table 5.

Type	**Topics**
Economic	stock market crash, recession, financial market, aviation industry, food industry, meat industry, restaurant industry, retail, tourism
Culture	cinema, education, sports, television, arts, music, fashion, performing arts, video games industry
Society	religion, gender, human rights, healthcare workers, strikes, social, mental health, racism, public transport

Table 1: Topics for the pandemic impacts context used by the IR system.

Note that health and political impacts have been omitted. Political impacts are very hard to capture by a general frame and, according to the Wikipedia list, are most easiest captured by the means of territorials and country borders. Health impacts, too, need a proper investigation to define clear topics. Both topics likely require expert knowledge. A further evaluation is beyond the scope of this thesis and is therefore omitted. Pandemic impacts form a good candidates

for expressing possible user interests. But, due to time constraints experiments are not conducted with them.

Pandemic Mitigation Measures In order to mitigate pandemic impacts (i.e., "flatten the curve"), mitigation measures are conducted. Again, a Wikipedia article, namely *COVID-19 pandemic* serves as knowledge base (Wikipedia contributors, 2020b). Topics have been handpicked and are comprised of social distancing, self-isolation, respiratory hygiene protection (mouth-nose-coverings), hand washing, circulation and air filtration and surface cleaning. This work specifically performs experiments with the context of mitigation measures and its realizations hand washing, social distancing, air circulation and surface cleaning.

Utilizing Wikipedia articles has various practical reasons. Using knowledge sources of this type are common practice amongst the natural language processing community. The handling of such is an effective and widely accepted means. It allows to semi-automatically retrieve content, can be used to scale applications and allows to augment systems with knowledge bases (Banerjee, Ramanathan, & Gupta, 2007; Chen, Fisch, Weston, & Bordes, 2017; Strube & Ponzetto, 2006).

A hierarchical taxonomy of the previous properties can found in Figure 15. Note that the description can be seen as a low-level expansion of the *pandemic node* in Figure 14.

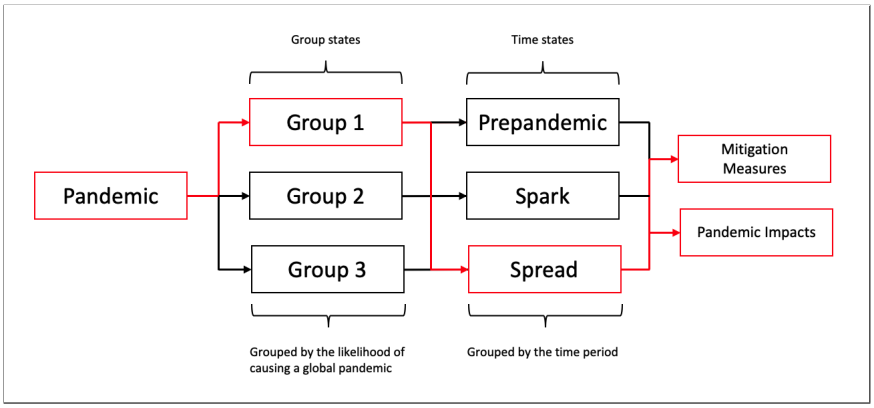

Figure 15: The pandemic taxonomy. The path highlighted in red describes the current COVID-19 pandemic. Note that at every point of time a pandemic is in exactly one group state and exactly one time state.

5 Context-Enhanced Information Retrieval

Two people performing search and entering the same query should not yield the same results. Understanding the user's context is a substantial part in comprehending what information to present. Specially the *topic of interest*, the social *role* and the *location of interest* form a context around the user in a pandemic (see Section 4). A system which is able to infer these information and presents tailored results would significantly increase performance of information retrieval (IR) systems (Ruthven, 2011). Consider the following two examples supporting this claim.

Example 1 This scenario plays in the early days of the COVID-19 pandemic. Both a child carer and a chief physician at a local hospital search for *effectiveness of face masks* in everyday life. The physician has a substantial education in hygiene and possibly virology. The child carer's single point of contact with the topic is the annual influenza epidemic. The physician wants to evaluate whether wearing face masks should be integrated in a holistic hygiene concept at the hospital. The child carer wants to know whether their kids should wear masks too. Though both searches may be the same, results should not.

Example 2 A mother wants to know whether their child should wear a face mask at school or not. She enters the query *'Should my child wear a face mask at school?'* into an IR system. The system retrieves the top hits. Likely, the hits are good but they may not be the best out-of-the-box. This can be explained as especially corona related articles are mostly comprised of their respective topic. Latent information such as the location or the people in the article are either hard to capture or do not influence the main topic enough. In this case, retrieval is mainly influenced by the main topic, namely face masks. Specially dynamic situations like pandemics are highly sensitive to their context like the location and people involved (Gladwell, 2002). The re-ranking method shown below tries to find this seemingly unimportant hidden information.

Ruthven (2011) precisely frames the high-level objective of the IR system shown in this section. We want to infer the user context upon the documents they explicitly select. The context is used to inform the IR system and better search results are iteratively presented throughout the session.

As shown in Section 4, the contexts used in this thesis are *social role*, *location of interest* and *topic of interest*. Section 5.2 explores how to represent the context. Experiments are conducted with the *topic of interest*, specifically *mitigation measures*. In Section 5.3 we assume the user context is known and show how this context may increase search accuracy. Experiments are conducted with the *location of interest*. Section 5.4 provides a strategy to infer the context of the user. Again, experiments are conducted with the *mitigation measures* and the *location of interest*.

5.1 System Specification

Model The IR system used in this thesis is based on a vector space model (see Section 3.4) enhanced by neural word embeddings (see Sections 2 and 3.5). Documents which comprise the same topics are close to each other. Documents dealing with different topics are very far away from each other.

Retrieval Process During run time, the user query will be embedded in the same vector space as the documents. Retrieving and ranking documents is conducted by the means of similarity. The closest documents to the query will be retrieved in that very order. This procedure is becoming common practice in neural IR (Mitra & Craswell, 2018).

Embeddings Documents, the query and contexts are all embedded in space using DistilBERT, a faster and lighter version of the BERT model (Sanh, Debut, Chaumond, & Wolf, 2020). For a word, the mean of the (often four) last layers is taken to create a embedding representation. To create sentence embeddings of arbitrary length, Sentence-BERT (SBERT) is being used (Reimers & Gurevych, 2019). The authors of SBERT suggest using DistilBERT as it is highly scalable and provided the best results in their test runs. SBERT simply adds a pooling operation to the output of the BERT model. This results in a mean representation of all words in a sentence. Semantically meaningful representations are computed and can be geometrically compared.

Similarity Sentence embeddings are vectors of real numbers. They are semantically ordered in space, therefore we can perform several operations to retrieve similarity. The most common operation is the cosine similarity, where $sim(a, b) = (a \cdot b|)/(|a| \cdot |b|)$. Unlike the dot product, the cosine similarity is a measure of direction only. This is because the denominator acts like a normalizing operation. When embeddings are compared results are in the range of $[0, 1]$, where 0 means absolutely no similarity and 1 means the exact same concept.

Context The context describes the surroundings of users. It is supposed to inform the IR about their preferences. Inferring the context is an essential part of the system, as we do not want to ask the user explicitly about their intent. Which contexts are relevant for this IR is discussed in Section 4.

A context is comprised of a set of topic instances. The context takes exactly one topic instance as a realizing state. E.g., in the following experiments, the context *mitigation measures* is comprised of the instances *face mask*, *social distancing*, *hand washing*, *surface cleaning* and *air circulation*. Assume that we know the user is interested in *face masks*. *Face masks* now realizes the user's context interest in *mitigation measures*. The objective will be to reorder retrieved documents by this context instance.

Re-ranking We now treat this context like a query. The context is embedded in the vector space and the previously retrieved documents are re-ordered according to the context.

5.2 Context Modelling

Assume a user is interested in the effectiveness of *face masks*. They enter their query and retrieve a list of documents. Most of the documents will be good, but as they go down the list, results will get increasingly worse (like a Google search). Documents which are bad may be relatively high up and documents which are good may be far down the list. Since we assumed to know the user's interest we can take the retrieved documents and find a new, better order. The new order shall be solely governed by the user's context. If a document is very similar to *face mask* it means high relevance for the user. If not, the document should be ranked low.

The context is supposed to capture the implicit intent of the user. Documents can be reordered according to their similarity to the context. In short, embedding the context in the vector space and reordering documents with respect to the topic *face mask* might increase the accuracy of the search.

For that, it is very important that the context representation is precise. If it is not, reordering according to this context will be imprecise too. The embedded context should be a meaningful representation of its concept. Meaningful means that, e.g., a context representation of *face mask* should be close to documents which comprise the very same topic. In principle, the objective can be understood like an (explicit) topic modelling task.

The main question arising at this point is how to model, or represent, the context such that it is as informative as possible. As we chose a neural vector space approach it is an obvious choice to experiment with various textual representations of the context. The idea is to explore candidates ranging from rich and broad to very short and concise representations. The textual representations we try are **1.** paragraphs from Wikipedia articles, **2.** short and concise summaries of these paragraphs, and **3.** the key-terms describing the respective concept.

Wikipedia articles are often written by experts or people who occupy themselves with the topic thoroughly. The articles include a large range of information including a brief description, synonyms, hypo- and hypernyms, definitions and related concepts. A Wikipedia article representation for *face mask* could be the introduction to cloth face mask (Wikipedia contributors, 2020a).

> *A cloth face mask is a mask made of common textiles, usually cotton, worn over the mouth and nose. When more effective masks are not available, and when physical distancing is impossible, cloth face masks are recommended by public health agencies for disease "source control" in epidemic situations to protect others from virus laden droplets in infected mask wearers' breath, coughs, and sneezes. Because they are less effective than N95 masks, surgical masks, or*

> *physical distancing in protecting the wearer against viruses, they are not considered to be personal protective equipment by public health agencies. They are used by the general public in household and community settings as perceived protection against both infectious diseases and particulate air pollution.*

The article includes (quasi-)synonyms and hyponyms like N95 mask, surgical masks or personal protective equipment and related concepts like breathing, sneezing, and coughing. The scope of a *cloth mask*'s application is described and their purpose is elaborated. Though the articles comprise a lot of information they yield a few issues. Long articles might become blurry in their semantic representation. Words with little semantic meaning like *the* or *and* might contribute to inaccuracies. Besides, they contain topics not necessarily describing the current object of observation, e.g., noisy words like *physical distancing* or *air pollution*. They are a potentially good candidate for a good contextual representation, but the question rises whether these deficits are problematic.

Summaries of the Wikipedia articles are supposed to make up for these deficits (if they reveal to be true). The brief description and synonyms shall be preserved. Other topics and words with little semantic meaning are removed. An example for a hand-crafted summary of the article above could be

> *A cloth face mask is made of common textiles worn over the mouth and nose to prevent viral infections with diseases.*

The summary solely talks about the *cloth face mask*. It is much more concise than the article. It tries to make up for the (presumably) noisy context of the Wikipedia by narrowing down content to the core. Then again, a lot of contextual information is lost.

Key-terms are the other extreme. Their position in the semantic space is exclusively governed by the terms itself. They describe a topic to the point and entail no noise at all. Though no synonyms or related concepts are contained they should, by the definition of word embeddings, contain their geometrically relatedness to other concepts. Key-terms, however, yield a major problem. Most word embeddings available were trained on pre-corona data. A concept like *face mask* has a low occurrence in the training data. The word embedding model simply does not know where to position the concept.

Experiment 1 The objective for the following experiments is to find which of the three introduced classes are most suitable for a context representation. Experiments are conducted for the context *mitigation measures*. The context contains four concepts, namely *hand washing*, *social distancing*, *face masks* and *air circulation*.

For each of the concepts we choose four hand-picked documents. Group one contains four articles related to *hand washing*, Group two contains four articles related to *social distancing*, and so on. Each group is now compared with possible context representations of the same topic.

For example, the documents in group one have the topic *hand washing*. They are compared with the three Wikipedia articles *Face masks during the COVID-19 pandemic*, *Surgical mask* and *Cloth face mask*. Next, the documents are compared with summaries of the three Wikipedia articles. Finally, a comparison between documents and key-terms is conducted. For each of the three textual representation class we also take the mean (denoted as M). The mean may make up for inaccuracies or biases by smoothing the representations. In short, for each topic, we compare four documents with $3 \cdot (3 + 1) = 12$ contextual representations.

The documents were hand-picked. The topics they comprise are distinct and describe circumstances precisely. Representations which have high similarity to those documents are likely good representations, as they are able to capture this topic well.

The respective documents chosen can be investigated in Table 5. In Figure 16 the similarity matrices for each of the four topics is presented.

		Hand Washing						Social Distancing						Face Masks						Air Circulation					
	No.	1	2	3	4	SA	OA	1	2	3	4	SA	OA	1	2	3	4	SA	OA	1	2	3	4	SA	OA
Wikipedia Articles	1	0,76	0,63	0,71	0,73	0,71	0,68	0,27	0,38	0,58	0,22	0,36	0,43	0,57	0,58	0,42	0,41	0,5	0,55	0,65	0,45	0,07	0,5	0,42	0,33
	2	0,65	0,6	0,61	0,6	0,62		0,39	0,42	0,65	0,27	0,43		0,67	0,63	0,51	0,49	0,58		0,59	0,21	0	0,36	0,29	
	3	0,73	0,63	0,61	0,65	0,66		0,38	0,5	0,66	0,35	0,47		0,64	0,63	0,46	0,38	0,53		0,41	0,2	-0	0,39	0,24	
	M	0,81	0,7	0,73	0,75	0,75		0,38	0,47	0,68	0,3	0,46		0,69	0,68	0,51	0,47	0,59		0,64	0,31	0,01	0,48	0,36	
Summaries	1	0,7	0,58	0,64	0,69	0,65	0,63	0,07	0,17	0,4	0,1	0,19	0,25	0,42	0,49	0,32	0,29	0,38	0,4	0,68	0,29	0,02	0,49	0,37	0,32
	2	0,57	0,53	0,59	0,58	0,57		0,16	0,23	0,46	0,06	0,23		0,43	0,47	0,36	0,27	0,38		0,63	0,22	0,03	0,38	0,32	
	3	0,67	0,59	0,65	0,64	0,64		0,27	0,31	0,45	0,14	0,29		0,51	0,44	0,34	0,25	0,38		0,42	0,2	-0	0,43	0,25	
	M	0,69	0,61	0,67	0,68	0,66		0,2	0,28	0,51	0,12	0,28		0,52	0,53	0,39	0,31	0,44		0,66	0,27	0,01	0,49	0,36	
Key-terms	1	0,58	0,47	0,54	0,58	0,54	0,52	0,01	0,1	0,25	-0	0,08	0	0,09	0,09	0,03	-0,1	0,02	0,09	0,1	-0,1	-0,2	0,22	0,01	0,09
	2	0,54	0,45	0,55	0,52	0,51		-0,1	-0,1	-0	0	-0,1		0,17	0,17	0,06	-0,1	0,09		0,26	-0	-0,2	0,23	0,07	
	3	0,54	0,4	0,47	0,55	0,49		-0,1	0,01	0,12	-0,1	-0		0,25	0,21	0,18	-0	0,16		0,33	0,08	-0,1	0,38	0,17	
	M	0,59	0,47	0,55	0,59	0,55		-0,1	0,01	0,13	-0,1	0		0,19	0,18	0,1	-0,1	0,1		0,25	-0	-0,2	0,3	0,09	

Figure 16: Comparing documents (columns) with Wikipedia articles, article summaries and key-terms (rows). SA = single average, OA = overall average, M = mean representation

The large 4×4 squares show the similarity of topics to the possible representation classes Wikipedia articles, summaries and key-terms. On the right of every square one can see the single average (SA) of single context representations and the overall average (OA) for the entire class. Values close to 1 mean a high similarity, whereas values close to 0 denote no similarity at all.

There are a few insights and interpretations to take from the experiments. For all four topics Wikipedia articles are the best representation. They have the highest overall average score. It is notable that summaries are not far off. Fine-tuning textual data with maximal information and minimal noise might be a feasible approach to obtain even better results. Possibly the summaries in

use were not accurate enough. Key-terms perform very badly. They likely do not capture the complexity of a topic very well.

Some documents are closer to *all* representations, hence the darker vertical lines. This means, these documents are easier to capture all concepts. The same same holds for light vertical lines, implying that some documents may be hard to capture. We conclude that rich representations seem to work well for the explicit topic modelling task at hand.

To make sure the results are not of random nature and *any* Wikipedia article is close to *any* topic we conduct a follow up experiment.

Experiment 2 The objective for the following experiments is to verify that Wikipedia articles are only close to documents with the same topic.

We use the exact same documents as before. The documents from group one (*hand washing* documents) are now compared to the four concepts *hand washing*, *social distancing*, *face masks* and *air circulation*. The concepts are all represented by the same Wikipedia articles. The same procedure is conducted for group two, and so on. In short, *all* documents are compared with *all* four contextual representations. Results can be seen in Figure 17.

			Documents															
			Hand Washing				Social Distancing				Face Masks				Air Circulation			
			1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Wikipedia Articles	Hand Washing	1	0,96	0,92	0,97	1	0,48	0,44	0,31	0,32	0,57	0,36	0,39	0,57	0,61	0,42	0,4	0,41
		2	0,82	0,88	0,84	0,82	0,71	0,62	0,58	0,68	0,71	0,61	0,55	0,81	0,79	0,67	1	0,52
		3	0,96	0,95	0,93	0,89	0,96	0,79	0,68	0,61	0,81	0,6	0,68	0,97	0,83	0,79	0,71	0,71
		M	1	1	1	0,99	0,78	0,67	0,56	0,58	0,76	0,57	0,59	0,85	0,81	0,68	0,76	0,59
	Social Distancing	1	0,48	0,55	0,65	0,48	0,69	0,76	0,85	0,61	0,61	0,77	0,65	0,77	0,8	0,73	0,34	0,88
		2	0,45	0,56	0,55	0,43	1	0,84	0,96	0,77	0,6	0,67	0,53	0,69	0,73	0,88	0,68	0,75
		3	0,65	0,7	0,79	0,56	0,97	1	0,97	1	0,69	0,85	0,8	0,94	1	0,86	0,66	0,99
		M	0,57	0,65	0,71	0,53	0,96	0,94	1	0,86	0,68	0,83	0,71	0,86	0,91	0,89	0,6	0,94
	Face Masks	1	0,58	0,63	0,64	0,59	0,89	0,69	0,74	0,7	0,82	0,85	0,82	0,72	0,85	0,86	0,45	0,85
		2	0,65	0,7	0,62	0,72	0,67	0,66	0,54	0,58	0,97	0,93	1	0,85	0,85	1	0,79	0,73
		3	0,46	0,49	0,52	0,43	0,45	0,44	0,55	0,44	0,92	0,93	0,89	0,66	0,69	0,75	0,39	0,92
		M	0,62	0,67	0,66	0,64	0,74	0,66	0,68	0,63	1	1	1	0,82	0,88	0,96	0,6	0,92
	Air Circulation	1	0,59	0,7	0,65	0,58	0,75	0,68	0,53	0,65	0,68	0,6	0,55	1	0,88	0,78	0,25	1
		2	0,31	0,38	0,48	0,34	0,58	0,32	0,13	0,56	0,52	0,26	0,29	0,83	0,89	0,37	0	0,72
		3	0,32	0,42	0,54	0,4	0,08	0,25	0,23	0,08	0,48	0,33	0,24	0,53	0,61	0,37	-0,1	0,79
		M	0,47	0,58	0,64	0,51	0,54	0,48	0,34	0,5	0,65	0,45	0,42	0,91	0,92	0,58	0,05	0,97

Figure 17: Comparing all documents (columns) with all Wikipedia articles (rows) of all four topic instances. Matrix columns are normalized.

We can observe that the diagonal of 4×4 matrices in large matrix is colored significantly darker than the rest. This means that, indeed Wikipedia articles are closer to the respective document with the same topic.

For every document the position of the 1 denotes the mapping to one of the classes. The matrix is normalized with respect to the columns, hence the seemingly 'higher' similarity. The topics *hand washing*, *social distancing* and *face masks* are comparably easy to capture. Most of their documents are classified correctly.

Documents from *air circulation* are mostly classified incorrectly. Either the topic is hard to capture or the documents chosen are not as distinct as expected. Document three may be evidence for the latter. It is not close to any class (it looks like it due to the normalization, but it is not). The content of the document comprises a chapter for *all* classes, hence the uncertainty in classifying the document. This implies that documents with non-distinct topic borders to be problematic for more sophisticated systems. On the other hand, they may not even be important as we expect to user to interesting in a vertical search.

Interestingly to note is the intra-class quality of the Wikipedia articles. If one compares the *hand washing* Wikipedia article 1 with articles 2 and 3 we can see that article 1 fulfills the job better. It is closest to its own topic and furthest away from the other topics. Articles 2 and 3 are close to documents of the same topic, but also quite close to the documents of other topics. That leads to framing an objective to finding even better representations. Representations should maximize closeness to documents of their own topic and minimize closeness to documents of other topic.

It is surprising to see how well Wikipedia articles capture topics. One could reason that for very large texts, topics become blurry and meanings vanish. Apparently it is the other way around. The more information is embedded, the better semantic complexities are captured. Remarkably, the word embeddings in use do not even know what COVID-19 is. They were trained on pre-corona data. This may be one of the main factors for rich representations to perform well.

Due to time constraints, a deeper analysis cannot be conducted. But the results shown certainly give rise for more extended experiments and thorough explanations.

5.3 Improving Search with Context

In the previous section we explored strategies to represent context. Next, we explore how (or if) context can increase accuracy of retrieval results. In the frame of the COVID-19 pandemic we concluded that especially the *topic of interest*, the social *role* and the *location of interest* form an informing context around the user (see Section 4). For the following experiment we move on to the context of *location of interest* and leave the *topic of interest* behind.

We construct an artificial person with a very specific information need. Their information need is assumed to be known. In Section 5.2 we showed that this need can be expressed in the form of a contextual representations. We provide the user with a small toy corpus, where documents are either relevant (denoted as 0) or not relevant (denoted as 1) to the user. If enough documents of relevance are retrieved one can conclude that the user has closed their information gap.

Next, the IR system (as specified in Section 5.1) is presented to the user. They can now formulate a query which represents their need. If the user is not satisfied with the search result (as in the real world) they can adapt the query and get more specific. Depending on the query they enter, an ordered list of documents is presented. We investigate how relevant the retrieved documents are to the user. Assuming that the result will not be optimal, we use the user's known context to re-rank the document list. Again, relevance of the retrieved documents is investigated. The claim for this section is that the second relevance computation, namely the one with re-ranking, scores higher than the first one.

In summary, the user enters a search query into the system, the system retrieves the most relevant documents and re-ranks them according to the user's known context. Finally, a comparison between results with and without re-ranking is consulted.

Experiment 3 For this experiment we use example two from the introduction of Section 5. A mother wants to know whether their child should wear a face mask. Her location of interest can be described as *school*. She uses the IR system provided and queries the corpus. The documents retrieved are not very accurate, hence an adjustment to her initial query is made. At first, results without re-ranking are presented to her which are undoubtedly suboptimal. The second time, re-ranked documents are presented to her yielding much higher information value. The exact setting is defined as following.

The toy corpus in use is comprised of 9 documents (one got deleted after experiments were conducted). The documents can be seen in Table 2. Document relevances are annotated by hand in a binary manner (denoted by **R**). All documents chosen have the topic *face mask* except for document 7. Additionally, documents 1, 8, 7 and 9 comprise the topic *school*/*children*. The ideal ranking for the mother therefore would be a permutation of 1, 8 and 9 followed by arbitrary documents. This is, because only those documents comprise both topics *face mask* and *school*. An ideal sequence could be expressed as $[1, 8, 9, 2, 3]$ where its corresponding relevance sequence would be $[1, 1, 1, 0, 0]$.

The mother queries the corpus twice. The first time she composes a very

No.	Title	R	Link
1	'This Year's Must-Have Back-to-School Item: Masks for Children'	1	https://nyti.ms/33TAPWN
2	'Why Aren't Face Shields More Popular in California?'	0	https://nyti.ms/3mN8A4T
3	'Masks May Reduce Viral Dose, Some Experts Say'	0	https://nyti.ms/2RTTMmT
4	'Coronavirus: Which Mask Should You Wear?'	0	https://nyti.ms/3i14jaq
5	'You're Getting Used to Masks. Will You Wear a Face Shield?'	0	https://nyti.ms/3kCmXXQ
6	'We'll Be Wearing Masks for a While. Why Not Make Them Nice?'	0	https://nyti.ms/3cuXIDK
7	'For Mom in South Korea, Sending a Child Back to School Was Worth the Risk'	0	https://nyti.ms/330UYey
8	'How to Help Kids Embrace Mask-Wearing'	1	https://nyti.ms/2G79PL8
9	'Should Young Children Wear Masks?'	1	https://nyti.ms/3kOuHG3

Table 2: Toy corpus comprised of documents with topics *face mask* and/or *school*.

general query. She enters *'Should my child wear a face mask?'*. She assumes that the IR system will understand that children spend a lot of time in schools. Current IR systems, however, cannot reason complex relationships like this. The retrieval function as specified in 5.1 returns the documents $[4, 5, 1, 8, 9]$ with its relevance sequence $[0, 0, 1, 1, 1]$. Indeed the three documents of interest $1, 8, 9$ are returned, though, not ranked very high.

The mother notices that the all results indeed deal with *face masks*, but top results do not talk about her *location of interest school*. Consequently, the mother enters a much more **specific query** *'Should my child wear a face mask at school?'*. The retrieval function returns $[4, 1, 5, 8, 9]$ with its relevance sequence $[0, 1, 0, 1, 1]$. The ranking improves a little more, but still, the best results are in the last spots. For clarity, Table 3 shows the ideal ranking, compared with the general query ranking and the specific query ranking.

Ideal Ranking		General Ranking		Specific Ranking	
No.	**R**	**No.**	**R**	**No.**	**R**
1	1	4	0	4	0
8	1	5	0	1	1
9	1	1	1	5	0
4	0	8	1	8	1
5	0	9	1	9	1

Table 3: The ideal ranking yields information value 1. The general query ranking yields information value 0,6183. The specific query ranking yields information value 0,6798. Metric in use is nDCG@5.

Now assume the IR system is aware of the mother's context. It understands that her *location of interest* can be generally framed as *school*. In the first step, the query (no matter which) picks up list of documents not quite relevant to the mother. Now that her context is known, the IR system re-ranks the five retrieved documents using the concept of *school*. Documents closest to the known context are now the new indicator for the relevance ranking. In Figure 18 one can see this procedure.

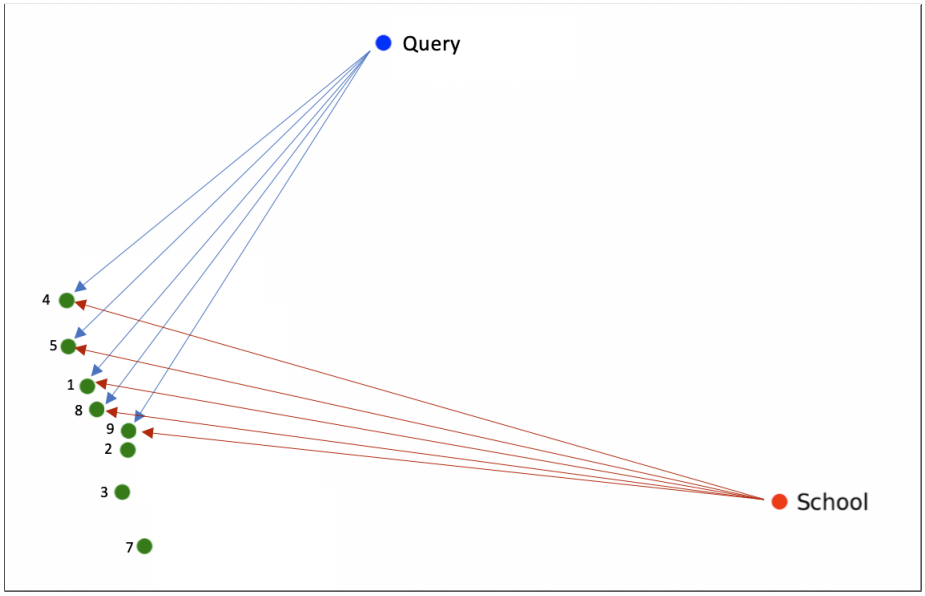

Figure 18: 2-dimensional PCA plot. The *query* picks up the 5 closest (in the order of smallest angle). Next, *school* re-ranks the documents.

Documents before re-ranking steps can be seen like unordered sets. The the previous order is completely given up for the new re-ranking order. Since the set for both query results comprise the same documents, re-ranking for both queries results in the **ideal ranking**.

5.4 Context Optimization

In the previous section we showed that given a user's context we can effectively re-rank search results according to the context. Unfortunately, assuming user's context is known raises a major issue. A person using an IR system for the first time could take any *role*, could be interested in any *topic* and any *location*. As a matter of fact, we have no prior information about the person using the system. It becomes evident very quickly that assuming the user's context is given, is a very simplified scenario.

A popular method for gaining information about users has been used for decades in an explicit manner. This procedure is known to everyone as *filtering* (Newell, 1997). Filtering is often conducted when structured information becomes available (Fischer & Stevens, 1991). These information could be metadata like authors, publisher or the date of an article. For unstructured data, simple filtering is not a valid method anymore. Data like newspaper contents could theoretically comprise an infinite number of topics, many of which are latent or simply hard to capture by explicit means.

In principle, the approach shown in Sections 5.2 and 5.3 acts like filtering. Though not pruning in a boolean manner (e.g., an article is written by an author or not), the context in use can be conducted to further narrow down search results. If the re-ranker returns less documents than the input amount, it acts like a fuzzy filter. Documents relevant according to the filter are ordered to top hits. At a certain threshold, documents not relevant enough are discarded, hence the fuzziness.

One feasible strategy to obtaining these information **explicitly** would be to ask additional questions on top of the user query. There are approaches which haven proven that only few questions are needed to make sense of an environment or context (Geman & Jedynak, 1993). A (quite sophisticated) example can be constructed as following. Assume a k-nery decision tree with depth n. Leafs describe a *role*, ordinary nodes are questions and edges are answers. With only n questions the most appropriate *role* out of k^n roles could be determined. Assuming a depth of $n = 3$ (questions) and a possibility of $k = 4$ answers for each question, only 3 questions are needed to determine the correct *role* out of $k^n = 4^3 = 64$ possibilities. Although a further exploration may be interesting to investigate, the exact procedure cannot be carried out and tested due to time constraints

There are certain circumstances where users will not want to reveal who they are. E.g., in **1.** time-constrained search, the user does not want to fine-tune the system until it fulfills their need. Or, due to **2.** privacy concerns, the user does want to provide possibly critical information. In this case, one possibility is to **implicitly** infer the user's interest by observing the documents they select. This observation and adaption procedure is often referred to as relevance feedback (Drucker et al., 2002).

In the following, we show a simple algorithm which infers the context of the user during their search process. The algorithm tries to optimize a variable amount of contexts (e.g., the contexts of *mitigation measures* and *location of*

interest). For each clicked document a searcher takes a step toward the right context instance until it converges (or gets lost in space). A thorough explanation including visualizations, an informal and formal definition is presented below.

The algorithm employs a simple heuristic, which assumes that the topic of a clicked document is the exact topic the user is interested in. Let us assume we want to find the two contexts *mitigation measures* and *location of interest*. We continuously observe the user's selected documents of interest and map the documents selected to the topics of the two contexts. A possible mapping could look like the one shown in Table 4.

Documents	Doc 1	Doc 2	Doc 3	Doc 4	···
mapped to	⇓	⇓	⇓	⇓	
Mitigation Measures	Hand Washing	Hand Washing	Hand Washing	Hand Washing	···
Location	School	School	Kinder-garten	School	···

Table 4: The observed documents are mapped to their respective context sequences.

For the *mitigation measure* the user's interest seems very clear. The user is mainly interested in *hand washing*. Their *location of interest*, however, is noisy. Though the user mainly selected documents with the location *school* they also clicked a document with the location *kindergarten*. For interpretability reasons relatively simple sequences were chosen. For each of the two context sequences, the respective context searcher is moved further to the topic of interest. Whichever topic the searcher is closest to by the end of an optimization step serves as the new context for the next query. In Figure 19 the procedure becomes clear.For the next iteration, the *query* picks up the five closest documents. The five documents serve as the new, pruned corpus. Next, the three documents closest to the previously found context *school* are kept. Eventually *hand washing* prunes the corpus to only one document. Pruning from five to three to one was chosen for illustration purposes.

A more general and a formal definition is given in the following.

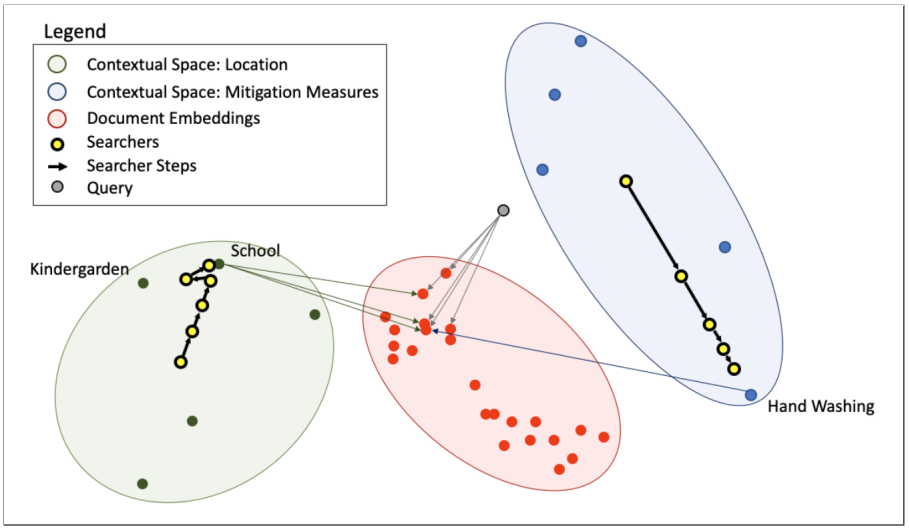

Figure 19: 2-dimensional PCA plot of the searcher convergence towards the topics *hand washing* and *school* in the respective searcher subspace.

Algorithm Assume a corpus of documents D comprised of n documents and a context C comprised of m topics. All are embedded in a vector space. A searcher S is placed in the middle of the topics. The aim of the algorithm is to move the searcher as close to one of the topics as possible. Whichever topic is closest to the searcher will be treated as the new context.

To accomplish this the sequence of documents clicked by the user is observed. Every document in this sequence is mapped to its closest topic. The underlying assumption is that a clicked document inhabits the exact topic the user is interested in. Doing this, we retrieve a sequence of topics. For each topic in the sequence, the searcher moves a certain distance toward this topic. Once finished, the topic closest to the searcher serves as the new context.

The optimization process is quite similar to the simulated annealing algorithm (Van Laarhoven & Aarts, 1987). In the beginning the searcher takes very large steps. Due to high uncertainty we allow adjustments in every direction. As we progress, we only take small steps, since we assume the initial direction taken was correct. Now we only need to search in a small subspace. At a certain point the searcher merely moves at all. We have found (if the claim of this algorithm is true) the optimal context. The formal algorithm is given below in Python-like pseudocode.

Formal Definition

```
def optimization_step(searcher, clicked_docs, context, n):
        """
        Performs an optimization of the searcher.
        :param searcher: The searcher vector.
        :param clicked_docs: List of document vectors.
        :param context: List of topic vectors of the context.
        :param n: The nth optimization step taken.
        """

        # Map the searcher to its closest topics
        topics = []
        for doc in clicked_docs:
            closest_topic = get_closest(context, searcher)
            topics.append(closest_topic)

        # Move the searcher in direction of topics clicked
        for topics in topics:
            direction = topic - searcher
            distance  = (1/1.3)**n / len(topics)
            searcher += direction * distance

        return searcher
```

Experiment 4 Again, we construct an artificial scenario to show the optimization steps proposed by the algorithm. A toy corpus of 20 documents consisting of five topics with four documents each is sourced. Documents' topics are annotated by hand. We assume that out of all possibilities the user is interested in *hand washing*. Documents in use can be seen in Table 5.

As described before, documents, the context (*mitigation measures* with its four instances) and a searcher are embedded in space. The user enters their query *'Is hand washing an effective measure against the virus?'*, which is embedded in space, too. The system returns the ordered list $[12, 11, 9, 5, 6, 17, 7, 10]$. Clearly, the retrieved documents are not optimal. 12, 11 and 9 are indeed the correct results as they are *hand washing* documents, but 10 is very far off.

We do, assume that the user will see the three documents relevant to them (documents 9, 10 and 11) and clicks them. In Figure 20 one can see that documents the three documents are indeed closest to *hand washing*. Therefore the searcher moves in the direction of *hand washing*. Note that due to dimensionality reduction ratios are not perfectly retained.

No.	Title	Topic	Link
1	'Relaxing the Rules of Social Distancing'	Social Distancing	https://nyti.ms/307T7mh
2	'Social Distancing Informants Have Their Eyes on you'	Social Distancing	https://nyti.ms/340hrrr
3	'Wondering About Social Distancing?'	Social Distancing	https://nyti.ms/340hjrX
4	'The Pandemic Isn't Over. New Yorkers Are [...].'	Social Distancing	https://nyti.ms/3mPyi8Y
5	'How to Clean Your Home for Coronavirus'	Surface Cleaning	https://nyti.ms/3kM3dRK
6	'Have I Been Cleaning All Wrong?'	Surface Cleaning	https://nyti.ms/3ctUoc8
7	'A Smarter Way to Clean Your Home'	Surface Cleaning	https://nyti.ms/2RVcXfX
8	'The Best Cleaners, Wipes, and Homemade [...]'	Surface Cleaning	https://nyti.ms/3626qs8
9	'How to Wash Your Hands'	Washing Hands	https://nyti.ms/3crxlyA
10	'The Hand-Washing Wars'	Washing Hands	https://nyti.ms/3i1Jg7E
11	'Four Swipes on Washing Your Hands'	Washing Hands	https://nyti.ms/33R42BO
12	'You've Been Washing Your Hands Wrong'	Washing Hands	https://nyti.ms/2FQE5dM
13	'How to Choose the Best Cloth Face Mask for You'	Face Masks	https://nyti.ms/33Rot1s
14	'A Detailed Map of Who Is Wearing Masks in the U.S.'	Face Masks	https://nyti.ms/3cuGyX4
15	'Why Aren't Face Shields More Popular in California?'	Face Masks	https://nyti.ms/3hU2Kek
16	'What Happens to Viral Particles on the Subway'	Face Masks	https://nyti.ms/340iM1r
17	'Opinion | Your Building Can Make You Sick [...]'	Air Filtration	https://nyti.ms/3i0OwZ8
18	'Masks May Reduce Viral Dose, Some Experts Say'	Air Filtration	https://nyti.ms/3ctVwfS
19	'Airborne Coronavirus: What You Should Do Now'	Air Filtration	https://nyti.ms/2HqKxs9
20	'Opinion | Yes, the Coronavirus Is in the Air'	Air Filtration	https://nyti.ms/35ZyvQS

Table 5: A toy corpus of 20 documents comprised of the five topics for the context *mitigation measures*.

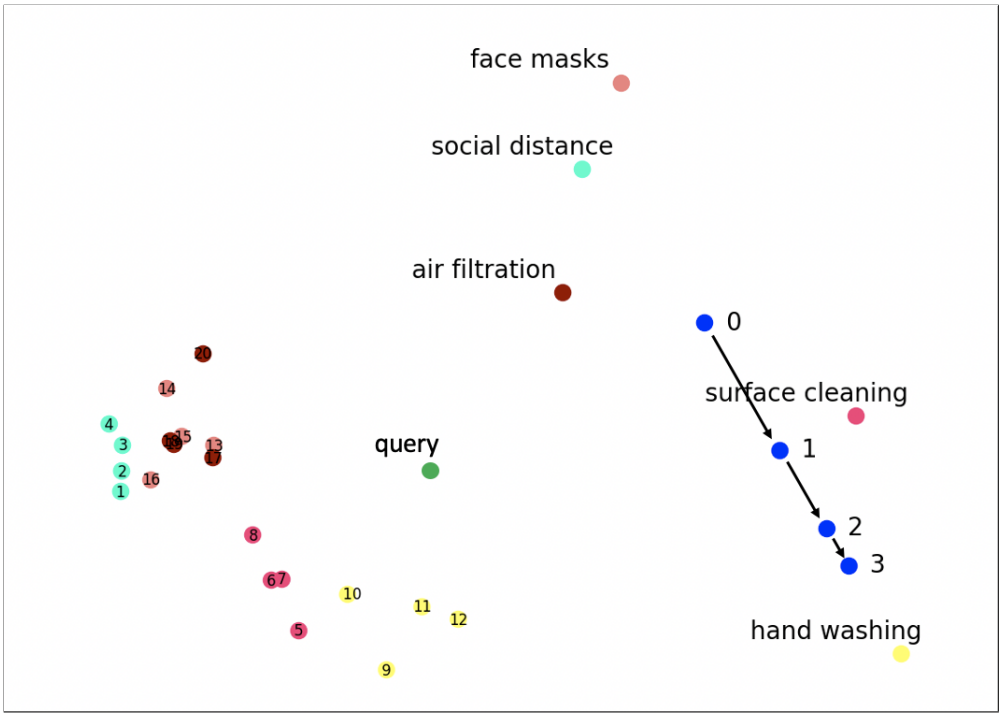

Figure 20: 2-dimensional PCA plot of the searcher convergence towards the topic *hand washing*.

5.5 Limitations

If the experiments shown were conducted "in the wild", without proper fine-tuning one would very likely encounter several problems. The experiments conducted evidently were very simplified such that one can interpret results and get a basic notion of the approach. Simplifications include the following issues and give rise to further investigation.

1. **Distinct Topics:** The documents selected were hand-picked such that it is clear which topic they belong to and where in the semantic space they sit. Conducting experiments with larger corpora are a reasonable next step.

2. **Noise:** The scenario above assumes that the user will make the right decision with minimal noise in order to fulfill their information need. Obviously the user will select topics which have little to do with his true topic of interest. Understanding which documents *really* contribute to the user's interest is of significant importance.

3. **Role Hopping:** We assumed that for the moment the user will remain in a very specific social role (see discussion in Section 4.2) until they

terminate the session. It would be interesting to investigate if there is a way to determine the type of role they inhabit and determine changes in the role spectrum of the respective user.

4. **Ranking Mixture:** Mitra and Craswell (2018) suggest to use a mix of neural and probabilistic methods. Due to time constraints, only neural methods were tested. It would be interesting to see a fine-tuned system utilizing both.

5. **Context Inference**: As discussed in Section 4 one might infer a person's *location of interest* if their *role* was known. In this approach this claim was examined. There is reason to believe the claim is true, but formalizing the problem seems like a very hard task.

6. **Embeddings**: The experiments conducted were solely based on DistilBERT embeddings. Other models yield other embeddings, surely with results better or worse. Trying other embeddings might be a reasonable step to fine-tune the system.

5.6 Conclusion

In this part of the thesis we showed that neural vector space models are able to capture and represent context quite effectively. Wikipedia articles revealed to be the best approach to represent the context. The articles are very rich in information and condense synonyms, hyponyms and related concepts. Neural word embeddings are able to capture the concept *very* effecively, even though COVID-19 is a completely unknown term. Also, noise and presumably blurring length were not a problem for the embeddings. Summaries showed some effectiveness. Considering they were handcrafted and articles were processed raw, one could consider fine-tuning the texts to get even better results. With a look on further research and advances in neural word embeddings an unsupervised, explicit topic modelling approach might be an approach to further investigate. Besides, we showed that a given a user's context the system is able to retrieve better search results tailored to the user's needs. Of course, assuming to know the user's context is a simplified scenario. Thereupon, a context inferring strategy was presented under simplified conditions.

References

Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender systems handbook* (pp. 217–253). Springer.

Atran, S. (1993). *Cognitive foundations of natural history: Towards an anthropology of science*. Cambridge University Press.

Ba, J., Mnih, V., & Kavukcuoglu, K. (2014). *Multiple object recognition with visual attention.* arxiv preprint 1412.7755.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). *Neural machine translation by jointly learning to align and translate.* arxiv preprint 1409.0473.

Banerjee, S., Ramanathan, K., & Gupta, A. (2007). Clustering short texts using wikipedia. In *Proceedings of the 30th annual international acm sigir conference on research and development in information retrieval* (pp. 787–788).

Barry, J. M. (2005). *The great influenza: the epic story of the deadliest plague in history*. Penguin Books.

Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, *3*, 1137–1155.

Bharat, K. (2000). Searchpad: Explicit capture of search context to support web search. *Computer Networks*, *33*, 493–501.

Boughareb, D., & Farah, N. (2014). Context in information retrieval..

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... Amodei, D. (2020). *Language models are few-shot learners.* arxiv preprint 2005.14165.

Bush, V. (1945). As We May Think. *Atlantic Monthly*, *176*, 641–649.

Cambridge Dictionary, A. (1999a). *Context*. Cambridge University Press. (Accessed on Sept., 21, 2020)

Cambridge Dictionary, A. (1999b). *Retrieval*. Cambridge University Press. (Accessed on Sept., 11, 2020)

Chen, D., Fisch, A., Weston, J., & Bordes, A. (2017). Reading wikipedia to answer open-domain questions.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). *Empirical evaluation of gated recurrent neural networks on sequence modeling.* arxiv preprint 1412.3555.

Coecke, B., Sadrzadeh, M., & Clark, S. (2010). *Mathematical foundations for a compositional distributional model of meaning.* arxiv preprint 1003.4394.

Crestani, F., & Pasi, G. (2000). *Soft computing in information retrieval*. Physica-Verlag Heidelberg.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). *Transformer-xl: Attentive language models beyond a fixed-length context.* arxiv preprint 1901.02860.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding.* arxiv preprint 1810.04805.

Drucker, H., Shahrary, B., & Gibbon, D. C. (2002). Support vector machines: relevance feedback and information retrieval. *Information processing &*

management, *38*(3), 305–323.

Ethayarajh, K. (2019). How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on world wide web* (pp. 406–414).

Fischer, G., & Stevens, C. (1991). Information access in complex, poorly structured information spaces. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 63–70).

Gauch, S., Chaffee, J., & Pretschner, A. (2003). Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems: An international Journal*, *1*(3), 219–234.

Geman, D., & Jedynak, B. (1993). Shape recognition and twenty questions.

Gladwell, M. (2002). *The tipping point: how little things can make a big difference.* Back Bay Books Boston.

Gong, Z., Cheang, C. W., & Hou, U. L. (2005). Web query expansion by wordnet. In *International conference on database and expert systems applications* (pp. 166–175).

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep residual learning for image recognition.* arxiv preprint 1512.03385.

Hiemstra, D. (2009). Information retrieval models. In A. Göker & J. Davies (Eds.), *Information retrieval: Searching in the 21st century* (pp. 1–19). Wiley.

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

Holmstrom, J. E. (1948). *Report and papers submitted*. Royal Society (Great Britain). Scientific Information Conference.

Hui, K., Yates, A., Berberich, K., & de Melo, G. (2017). *Pacrr: A position-aware neural ir model for relevance matching.* arxiv preprint 1704.03940.

Hutson, M. (2018). *Has artificial intelligence become alchemy?* (Vol. 360).

Ingwersen, P., & Järvelin, K. (2005). *The turn: Integration of information seeking and retrieval in context*. Springer.

Jonassen, D. (2003). Using cognitive tools to represent problems. *Journal of research on Technology in Education*, *35*(3), 362–381.

Ke, G., He, D., & Liu, T.-Y. (2020). *Rethinking positional encoding in language pre-training.* arxiv preprint 2006.15595.

Kim, Y., Denton, C., Hoang, L., & Rush, A. M. (2017). *Structured attention networks.* arxiv preprint 1702.00887.

Kraft, D. H., Bordogna, G., & Pasi, G. (1994). An extended fuzzy linguistic approach to generalize boolean information retrieval. *Information Sciences - Applications*, *2*(3), 119-134.

Kuchaiev, O., & Ginsburg, B. (2017). *Factorization tricks for lstm networks.* arxiv preprint 1703.10722.

Kuhlen, R. (1990). Zum stand pragmatischer forschung in der informationswis-

senschaft. In (pp. 13–18). Univ.-Verl. Konstanz.
Kuhlen, R. (2013). *Grundlagen der praktischen information und dokumentation: Handbuch zur einführung in die informationswissenschaft und -praxis* (No. 6). De Gruyter Saur.
Kuropka, D. (2004). *Modelle zur repräsentation natürlichsprachiger dokumente.* Logos-Verlag.
Lalmas, M. (2011). *Information retrieval summer school 2011.* (Accessed on Sept., 09, 2020)
Lee, J. H. (1994). Properties of extended boolean models in information retrieval. In *Sigir '94* (pp. 182–190). Springer London.
Li, F.-F., Johnson, J., & Yeung, S. (2017). *Cs231n, lecture 4.* Stanford University.
Liu, T.-Y. (2009). *Learning to rank for information retrieval.*
Luong, M.-T., Pham, H., & Manning, C. D. (2015). *Effective approaches to attention-based neural machine translation.* arxiv preprint 1508.04025.
Madhav, N., Oppenheim, B., Gallivan, M., Mulembakani, P., Rubin, E., & Wolfe, N. (2017). *Pandemics: Risks, impacts, and mitigation.* The International Bank for Reconstruction and Development / The World Bank.
Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval.* Cambridge University Press.
Merriam-Webster. (n.d.).
Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space.* arxiv preprint 1301.3781.
Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). *Distributed representations of words and phrases and their compositionality.* arxiv preprint 1310.4546.
Mitra, B., & Craswell, N. (2018). An introduction to neural information retrieval. *Foundations and Trends in Information Retrieval*, *13*(1), 1-126.
Mnih, V., Heess, N., Graves, A., & Kavukcuoglu, K. (2014). *Recurrent models of visual attention.* arxiv preprint 1406.6247.
Mylonas, P., Vallet, D., Castells, P., Fernández, M., & Avrithis, Y. (2008). Personalized information retrieval based on context and ontological knowledge. *The Knowledge Engineering Review*, *23*(1), 73–100.
Myrhaug, H. I., & Goker, A. (2003). Ambiesense-interactive information channels in the surroundings of the mobile user. In *Univeral access in hci, 10th international conference on human-computer interaction* (Vol. 4, pp. 1158–1162).
Newell, S. C. (1997). User models and filtering agents for improved internet information retrieval. *User Modeling and User-Adapted Interaction*, *7*(4), 223–237.
Nirenburg, S., McShane, M., Beale, S., Wood, P., Scassellati, B., Mangin, O., & Roncone, A. (2018). Toward human-like robot learning. In *Natural language processing and information systems* (Vol. 23, pp. 73–82). Springer.
Oard, D. W., Kim, J., et al. (1998). Implicit feedback for recommender systems. In *Proceedings of the aaai workshop on recommender systems* (Vol. 83).
Ono, M., Miwa, M., & Sasaki, Y. (2015). Word embedding-based antonym

detection using thesauri and distributional information. In *Proceedings of the 2015 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 984–989).

Parikh, A., Täckström, O., Das, D., & Uszkoreit, J. (2016). A decomposable attention model for natural language inference. In *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 2249–2255). Association for Computational Linguistics.

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543). Association for Computational Linguistics.

Perera, C., Zaslavsky, A., Christen, P., & Georgakopoulos, D. (2013). Context aware computing for the internet of things: A survey. *IEEE communications surveys & tutorials*, *16*(1), 414–454.

Peters, M. E., Ammar, W., Bhagavatula, C., & Power, R. (2017). *Semi-supervised sequence tagging with bidirectional language models.* arxiv preprint 1705.00108.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *CoRR*.

Preuveneers, D., Van den Bergh, J., Wagelaar, D., Georges, A., Rigole, P., Clerckx, T., ... De Bosschere, K. (2004). Towards an extensible context ontology for ambient intelligence. In *European symposium on ambient intelligence* (pp. 148–159). Springer.

Reimers, N., & Gurevych, I. (2019). *Sentence-bert: Sentence embeddings using siamese bert-networks.* arxiv preprint 1908.10084.

Robertson, S., & Zaragoza, H. (2009). *The probabilistic relevance framework: Bm25 and beyond*. Now Publishers Inc.

Ruthven, I. (2011). Information retrieval in context. In M. Melucci & R. Baeza-Yates (Eds.), *The information retrieval series* (pp. 187–207). Springer-Verlag Berlin Heidelberg.

Salton, G., Fox, E. A., & Wu, H. (1983). Extended boolean information retrieval. , *26*(11), 1022–1036.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, *18*(11), 613–620.

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2020). *Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.* arxiv preprint 1910.01108.

Schmidhuber, J., Bengio, Y., & Frasconi, P. (2003, 03). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. *A Field Guide to Dynamical Recurrent Neural Networks*.

Schramm, W. (1954). How communication works. *The process and effects of mass communication*, *3*, 26.

Shen, X., Tan, B., & Zhai, C. (2005). Context-sensitive information retrieval using implicit feedback. In *Proceedings of the 28th annual international acm sigir conference on research and development in information retrieval* (pp. 43–50).

Strube, M., & Ponzetto, S. P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *Aaai* (Vol. 6, pp. 1419–1424).

Tamine-Lechani, L., Boughanem, M., & Daoud, M. (2010). Evaluation of contextual information retrieval effectiveness: overview of issues and research. *Knowledge and Information Systems*, *24*(1), 1–34.

Tang, Y., Srivastava, N., & Salakhutdinov, R. (2013). *Learning generative models with visual attention.* arxiv preprint 1312.6110.

Vallet, D., Fernández, M., Castells, P., Mylonas, P., & Avrithis, Y. (2006). Personalized information retrieval in context. In *3rd international workshop on modeling and retrieval of context* (pp. 16–17).

Van Laarhoven, P. J., & Aarts, E. H. (1987). Simulated annealing. In *Simulated annealing: Theory and applications* (pp. 7–15). Springer.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *CoRR*.

Wang, L. L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Burdick, D., ... Kohlmeier, S. (2020). *Cord-19: The covid-19 open research dataset.* arxiv preprint 2004.10706.

White, R. W., Jose, J. M., & Ruthven, I. (2006). An implicit feedback approach for interactive information retrieval. *Information processing & management*, *42*(1), 166–190.

Wikipedia contributors. (2020a). *Cloth face mask.* (Accessed on Sept., 30, 2020)

Wikipedia contributors. (2020b). *Covid-19 pandemic.* (Accessed on Sept., 22, 2020)

Williamson, G. (n.d.). *The encode-decode model of communication.* (Accessed on Sept., 23, 2020)

Winston, P. (2014). The genesis story understanding and story telling system - a 21st century step toward artificial intelligence..

Wurm, C. (2018). *Neuronale netze und tiefe architekturen.* Retrieved from `https://user.phil.hhu.de/~cwurm/wp-content/uploads/2019/01/deep_architectures.pdf` (Accessed on Sept., 11, 2020)

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., ... Bengio, Y. (2015). *Show, attend and tell: Neural image caption generation with visual attention.* arxiv preprint 1502.03044.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). *Xlnet: Generalized autoregressive pretraining for language understanding.* arxiv preprint 1906.08237.