



Interdisciplinary Journal of E-Skills and Lifelong Learning

An Official Publication
of the Informing Science Institute
InformingScience.org

IJELL.org

Volume 16, 2020

FROM AN ARTIFICIAL NEURAL NETWORK TO TEACHING

Dror Mughaz *	Dept. of Computer Science, Bar-Ilan University, Ramat-Gan, Israel, and Dept. of Computer Science, Lev Academic Center, Jerusalem, Israel	myghaz@gmail.com
Michael Cohen	Dept. of Computer Science, Lev Academic Center, Jerusalem, Israel	michael.co24@gmail.com
Sagit Mejahez	Ministry of Education, Jerusalem, Israel	sagmej@gmail.com
Tal Ades	Dept. of Computer Science, Lev Academic Center, Jerusalem, Israel	talades8@gmail.com
Dan Bouhnik	Dept. of Computer Science, Lev Academic Center, Jerusalem, Israel, and Dept. of Information Science, Bar-Ilan University, Ramat-Gan, Israel	dan.bouhnik@biu.ac.il

ABSTRACT

Aim/Purpose	Using Artificial Intelligence with Deep Learning (DL) techniques, which mimic the action of the brain, to improve a student's grammar learning process. Finding the subject of a sentence using DL, and learning, by way of this computer field, to analyze human learning processes and mistakes. In addition, showing Artificial Intelligence learning processes, with and without a general overview of the problem that it is under examination. Applying the idea of the general perspective that the network gets on the sentences and deriving recommendations from this for teaching processes.
Background	We looked for common patterns of computer errors and human grammar mistakes. Also deducing the neural network's learning process, deriving conclusions, and applying concepts from this process to the process of human learning.
Methodology	We used DL technologies and research methods. After analysis, we built models from three types of complex neuronal networks – LSTM, Bi-LSTM, and GRU – with sequence-to-sequence architecture. After this, we combined the sequence-to-sequence architecture model with the attention mechanism that gives a general overview of the input that the network receives.

Accepted by Editor Fay Sudweeks | Received: February 19, 2020 | Revised: June 18, 2020 |
Accepted: June 19, 2020.

Cite as: Mughaz, D., Cohen, M., Mejahez, S., Ades, T., & Bouhnik, D. (2020). From an artificial neural network to teaching. *Interdisciplinary Journal of e-Skills and Lifelong Learning*, 16, 1-17. <https://doi.org/10.28945/4586>

(CC BY-NC 4.0) This article is licensed to you under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/). When you copy and redistribute this paper in full or in part, you need to provide proper attribution to it to ensure that others can later locate this work (and to ensure that others do not accuse you of plagiarism). You may (and we encourage you to) adapt, remix, transform, and build upon the material for any non-commercial purposes. This license does not permit you to use this material for commercial purposes.

Contribution	The cost of computer applications is cheaper than that of manual human effort, and the availability of a computer program is much greater than that of humans to perform the same task. Thus, using computer applications, we can get many desired examples of mistakes without having to pay humans to perform the same task. Understanding the mistakes of the machine can help us to understand the human mistakes, because the human brain is the model of the artificial neural network. This way, we can facilitate the student learning process by teaching students not to make mistakes that we have seen made by the artificial neural network. We hope that with the method we have developed, it will be easier for teachers to discover common mistakes in students' work before starting to teach them. In addition, we show that a "general explanation" of the issue under study can help the teaching and learning process.
Findings	We performed the test case on the Hebrew language. From the mistakes we received from the computerized neuronal networks model we built, we were able to classify common human errors. That is, we were able to find a correspondence between machine mistakes and student mistakes.
Recommendations for Practitioners	Use an artificial neural network to discover mistakes, and teach students not to make those mistakes. We recommend that before the teacher begins teaching a new topic, he or she gives a general explanation of the problems this topic deals with, and how to solve them.
Recommendations for Researchers	To use machines that simulate the learning processes of the human brain, and study if we can thus learn about human learning processes.
Impact on Society	When the computer makes the same mistakes as a human would, it is very easy to learn from those mistakes and improve the study process. The fact that machine and humans make similar mistakes is a valuable insight, especially in the field of education, Since we can generate and analyze computer system errors instead of doing a survey of humans (who make mistakes similar to those of the machine); the teaching process becomes cheaper and more efficient.
Future Research	We plan to create an automatic grammar-mistakes maker (for instance, by giving the artificial neural network only a tiny data-set to learn from) and ask the students to correct the errors made. In this way, the students will practice on the material in a focused manner. We plan to apply these techniques to other education subfields and, also, to non-educational fields. As far as we know, this is the first study to go in this direction – instead of looking at organisms and building machines, to look at machines and learn about organisms.
Keywords	deep-learning, text-mining, Hebrew, subject-tagger

INTRODUCTION

The world of Machine Learning (ML) is very large. ML is used in many areas such as: health, economics, web search engines, image processing, robotics, etc. It is customary to divide ML algorithms into several types: (1) "Supervised learning" (2020) – each example comes with a classification label. The purpose of the algorithm is to predict the classification of new examples that it did not encounter in the learning process. Artificial neural network training relies on such algorithms. (2) "Unsupervised learning" (2020) – the purpose of the algorithms is to find a simple representation or template for understanding the data collection. Common methods of this type are clustering and low-dimensional spreading such as principal factor analysis. (3) Reinforcement learning (Kaelbling et al., 1996) – the learning algorithm receives partial feedback on its performance (only after completion of the assignment) and must conclude which of its decisions led to success/failure. The motivation of this

research is how (if we can) to exploit the mistakes of a ML in favor of humans, i.e., to analyze the ML mistakes and teach humans not to make the same mistakes.

The goal of this study is to examine the learning process of computational tools built according to the human brain model and to learn from their mistakes. There are a number of ML algorithms; one of the common mathematical models in this field is the Artificial Neural Network (ANN, 2020) model. ANN, as used in ML, is a computational mathematical model based on the structure of the human brain or on the cognitive processes that take place in a natural neural network. The network consists of a number of “neurons” arranged in layers, with each neuron being able to interact with a number of other neurons in the system. Each neuron is capable of simple computational operations and, in turn, transmits the information, i.e., number, it deduced to other neurons. In this way, as data advances through the ANN layers, the system transforms the raw data into valuable, usable information. In order to “teach” the network how to avoid mistakes, we can use a feedback mechanism, known as a back-propagation algorithm. This mechanism enables the network to adjust the connections back through the network. By applying this algorithm, the network can go back and “double-check” to make sure that all the biases are correct and that all the connections are weighted correctly. As a result, the system learns to make more accurate decisions. One of the primary properties of the neural networks is their ability to emulate the brain’s pattern-recognition skills. Neural networks are used for a wide array of tasks such as predicting the outcome of investment decisions, finding patterns in handwriting, and even facial scanning to identify a person.

Use of ANN has become very widespread. The ANN model has great potential to provide important information to researchers in various domains, such as the academic, industrial, medical, and communications domains. Currently, computerized corpora are the basis of many textual projects; as a result, ANN is of prime importance in this field.

As far as we know, to this day, researchers in the AI field have not taken the approach of using ANN’s results to try to draw (technical) conclusions about humans. AI and DL researchers have focused on, and are still focusing on, the opposite direction only: studying how to enable the computer model to imitate the organism (humans). However, currently, researchers still can’t explain why the ANN model comes to one conclusion or another (even if the network’s conclusion is correct) even though, in recent years, researchers have begun to work intensively to try to explain why the ANN model made the decision it did. Additionally, when researchers currently try to solve problems using an ANN model, every once in a while they run into problems that an organism manages to solve and AI does not. To cope with this phenomenon, researchers try to improve the existing models, but today, even after great advances in the AI field, AI is still not close enough to human capabilities. However, this does not mean that it is not worthwhile to investigate the opposite direction – from the machine to the organism. This work is preliminary, and we are now performing the first experiments, as far as we know, in this direction.

We will examine the learning processes of a number of such models and try to draw conclusions about human learning processes, because the ANN model is based on the human brain. For example, if ANN makes some mistakes on a specific task, we might expect human students to make the same or similar mistakes. In this way, we can be better prepared for the teaching process. Beyond that, it is possible that by analyzing how ANN deals with a particular task we can estimate what decision a person would make when dealing with the same or a similar task.

One way to research this is to analyze ML errors. ML makes mistakes, and also humans make mistakes. If we analyze the mistakes that ML makes, maybe we can learn what mistakes humans will make. Then, we can teach humans not to “repeat” the mistakes that ML made. By doing so, we can improve the quality of human learning. While taking a human survey of a learning process, and choosing the right people to participate in it, is expensive and sometimes not easy, using a computer to model the learning process is much cheaper and simpler. To our knowledge, we are the first to investigate this area.

In this work, we will deal with the syntactical analysis of Hebrew language texts, especially Hebrew sentences. The task being taught is identifying the grammatical subject of a sentence. Our goal is to compare ANN mistakes in performance of this task to human mistakes, in order to identify and assess common human grammar mistakes and to analyze the ways humans can perform this task.

Semitic languages are quite dissimilar from Indo-European languages. Hebrew texts are processed differently than the English language because (1) texts written in Indo-European languages are written from left-to-right, while those written in Hebrew are written from right-to-left (Wintner, 2004); (2) in comparison to Indo-European languages such as English, German, and French, little Natural Language Processing (NLP) research has been done on the Hebrew language to date; (3) Hebrew is a morphologically rich language, e.g., words can have many prefix forms (“and when in ...”, “and when ...”, “and ...”, “when ...”, “in ...”). One of the results of this complex morphology is ambiguous words (HaCohen-Kerner, Kass & Peretz, 2010); (4) Hebrew texts contain many acronyms and abbreviations (HaCohen-Kerner et al., 2004). HaCohen-Kerner et al. (2004) show that there are 40,000 abbreviations in Hebrew, compared to 17,000 in English. A study made in 2013 (HaCohen-Kerner et al., 2013) shows that the manual disambiguation of an acronym is a very time-consuming process, and it is a very difficult task even for a professional; (5) Hebrew is an ancient language; people have been speaking that language since the beginning of the 2nd millennium BCE. In Hebrew, there are written texts from two thousand years ago and more, most of them rabbinical (religious) texts (Mughaz, 2003; Mughaz et al., 2019a, 2019b).

This article is organized as follows: we will give a review of previous works; we will introduce the data-set and its pre-processing; we will explain the experiments and the tools we used; we will analyze the results; we will show how the experiments we did can have a practical implications, and at the end we will make conclusions and future works.

RELATED WORKS

The fact that ML learn from examples that humans feed them has been widely researched and discussed. ML is a useful aid in many areas such as face recognition (Deng et al., 2019; Zangeneh et al., 2020), robotics (Levine et al., 2018; Vemula et al., 2018), text mining (Luque et al., 2019; Mughaz et al., 2015, 2019a, 2019b), natural language processing (Eger et al. 2019; Kulkarni & Shivananda, 2019; Young et al., 2018), and machine translation (Bahdanau et al., 2014; Luong et al., 2015; Wu et al., 2016). However, none of these studies researched the opposite direction, i.e., taking examples of machine results/answers and learning from the error/behavior of the machine in order to teach humans to avoid making the same mistakes.

Following the general introduction of ANN, which appears in the introduction section, we then elaborate on the ANN sequential model. Sequence modeling is widely used in text analysis, for example, predicting the word/letter that comes next in a sequential input (Sutskever et al., 2011). This task is accomplished by computing the probability of occurrence of several words in a particular sequence. In sequence modeling, the current output is dependent not only on the current input but also on the previous input. Unlike other ML tasks, in sequence modeling, the input and output length are not fixed.

The basic and classic ANN is a feedforward net. A feedforward network (Schmidhuber, 2015) feeds information straight through the net, while Recurrent Neural Networks (RNN) (Mozer, 1988) cycle information through a loop, and the inputs are called recurrent. Given one input which is a series of data through time, e.g., a sentence, the Feedforward network has no interest in time or serial reference in time, e.g., words, and the only input it considers is the current example to which it is exposed. Thus, Feedforward networks have “amnesia” in regards to previous stages of the timeline (steps of series) they remember only the current moment. On the other hand, RNNs take as their input not just the current input example, also what they processed previously in the series or in the time.

RNNs can even apply to images, which can be decomposed into a series of patches and treated as a sequence. What distinguishes RNN networks from other neural networks is that they take time and sequence into account. Since recurrent networks possess a specific type of memory, and memory is also part of the human condition, the RNN has a partial analogy to the human brain.

The main problem with RNNs occurs when the sequence is too long; the networks have trouble carrying information from earlier steps to later ones. So, when processing a big paragraph of text, RNN's may 'forget' important information from the beginning. This problem is known as the "vanishing or exploding gradient problem" (Bengio et al., 1994) when gradient values are used to update the weights of the RNN. The problem of vanishing gradient arises when the values shrink or expand along the time. If the values of the gradients becomes very small, they do not significantly contribute to learning. So, in RNN, layers with a small gradient update impede learning.

The RNN variation plus "Long Short-Term Memory" (LSTM) was first proposed in 1997 by two researchers from Germany, Ied Hochreiter and Jorn Schmidhuber (Hochreiter & Schmidhuber, 1997). In their research, they show that LSTM units solve the RNN problem by preserving the error value over time and over layers. By maintaining the error value over time and over the RNN layers, the RNN can cause the learning process to continue over an extended period of time. The result was that LSTM has opened options for linking words that are far apart. Thus, in our research, we used LSTM.

The neuron network receives sentences/words as input and returns an output regarding the task it is coping with. Like all computational tools, neuronal networks cannot receive words as humans do; rather, they must receive the words numerically, i.e., vectors of numbers representing the words. The domain of converting words into representative vectors is called word embedding. There are several methods of embedding words, for example, one-hot encoding ("One-hot" 2020) or FT (Ramos, 2003); modern methods for embedding words include as word2vec (Mikolov, Chen et al., 2013; Mikolov, Sutskever et al., 2013) and glove (Pennington et al., 2014). The main idea of embedding words is that the words around a specific word define it; this is referred to as the Distributional Hypothesis. The Distributional Hypothesis is that if words are in a similar environment, then the semantic meaning of the words is similar (Harris, 1954). The main idea that "a word is characterized by the company it keeps" was proposed by Firth (1957). The Distributional Hypothesis is the basis for Statistical Semantics. Despite the fact that the Distributional Hypothesis originated in the study of Linguistics, it is now receiving attention in Cognitive Science (McDonald & Ramscar, 2001). The theoretical basis and source of the Distributional Hypothesis is discussed by Sahlgren (2008).

The RNN/LSTM/GRU network receives an input of vectors (each vector represents a word) and sequentially transmits the information contained in the vectors up to the end of the network. The information coming to the end of the network from the first vectors is less than the information coming from the last vectors. In many cases, there are vectors (words) at the beginning of the sentence that are very important for the last vectors (end of the sentence); however, in the serial transitions the information contained within them fades by the end of the sentence. For this purpose, (among other things) the attention mechanism (Bahdanau et al., 2014; Luong et al., 2015) is built. The attention mechanism takes all the vectors (all the sentence) as input and decides which of the representative's vectors (among the words in the sentence) is most important. In fact, in addition to the serial view of the vectors (the words in the sentence), the attention mechanism takes an overall view of all the vectors (sentence). This mechanism works well and improves performance.

Syntactic parsing is the task of building a syntactic parse tree of a sentence. This syntactic tree represents the structure of the sentence. The subject of a sentence is one of the properties that sentence syntactic parsing tries to uncover. Sentiment analysis is one type of task for which sentence parsing has been useful. Gómez-Rodríguez et al. (2019) empirically examined how important the quality of the syntactic analysis is for sentiment analysis, specifically polarity classification on English sentences as the target language. They evaluate their experiments using four well-known dependency parsers.

They concluded that better syntactically-labeled sentences do not necessarily lead to a significantly better accuracy in this task.

Jaf and Calder (2019) present a multi-lingual dependency parser using DL. The DL technique deals with common problems with parsing, for example, long-distance head attachment. One of the advanced DL techniques is transfer learning. Transfer learning exploits extensive knowledge of the resourced language and uses it for a limited-resourced language. Their study yielded interesting results of the effect of transfer learning on resource-limited languages, which always performed at the same, or a higher level than the best-known parsers.

Liebeskind and Liebeskind (2020) were interested in the task of classifying Hebrew historical texts according to their period of composition. Following the promising results of DL for various tasks in natural language processing (NLP), they used three DL models to deal with this problem, convolutional neural networks (CNN), LSTM, and GRU. The results of their experiments were that the GRU model reached an accuracy of 84.9%, a recall of 77.47%, and an F1 of 78.29%, and was better than the other models.

Our case study was performed on the Hebrew language. The number of studies performed on the Hebrew language is not large. Mughaz et al. (2018) classify short Hebrew texts according to the opinion of the writer. The corpus they used consists of short product reviews which were parsed into individual sentences. They applied the SVM algorithm on a combination of both unigrams and bigrams. Then they applied feature selection according to weights of the features by removing the features ranked less than 0.1. They tested the pruned features on SVM with a linear kernel and Bayesian Logistic Regression which yielded a success rate of 92.6% and 92.4%, respectively. Liebeskind (2019), and Liebeskind and Liebeskind (2019) extracted a Hebrew data-set of short user political comments. The aim was to predict the most emoji most closely corresponding to the text. They showed that word2vec Word Embedding is not optimal for this task; moreover, they showed that for the emoji prediction for political domain in Hebrew, the use of character n-grams representations exceeded all the other representation. Liebeskind et al. (2017) examined nine ML technics for classifying writer sentiment for a Hebrew Facebook corpus of 5.3 million messages. The Facebook messages were of incumbent politicians. They examined two different sentiment classification tasks, general attitude and attitude towards the content of the post. They combined two classes of features, Facebook-based and text-based features. They found that the n-grams character model text representation exceeded other representations. Their results showed that the Logistic Regression method exceeded the other eight ML models in terms of F-measures and accuracy.

Other studies that are related to document classification and address the challenges of Hebrew involve the classification of Hebrew-Aramaic documents according to style (Koppel et al., 2006; Mughaz, 2003); authorship verification, including forgeries and pseudonyms (Koppel et al., 2003, 2004) and classification of texts according to their ethnic origin and their historical period (HaCohen-Kerner, Beck, Yehudai & Mughaz, 2006; HaCohen-Kerner, Mughaz et al., 2008; HaCohen-Kerner, Beck, Yehudai, Rosenstein & Mughaz, 2010).

HaCohen-Kerner et al. (2011) used six ML techniques for identifying citations. To achieve this task, they used four feature types – n-gram, stop word-based, quantitative, and orthographic – and tested them separately and together. The best results were by combination of the four feature sets. Their study could identify if a sentence included a citation; it did not identify the citation itself.

Mughaz et al. (2015) extracted time-related key-phrases from rabbinical texts. They found that many of the sentences that hold time-related key-phrases also contain rabbinic names. They presented and applied a semi-automatic method that facilitates the extraction of time-related key-phrases. In other works of Mughaz et al. (2014a, 2014b, 2017, 2019a, 2019b) and HaCohen-Kerner & Mughaz (2010) they improved upon the previous method and used time-related phrases and references in order to date texts. The dating they suggested could help identify ancient anonymous texts and could even help identify edited texts.

In this study, we will use RNN in order to identify the subject of sentences. We will randomly select sentences that the RNN tagged incorrectly. Then, we will give the incorrectly tagged sentences to students, and we will examine how well the students succeeded in the subject-tagging task.

We do not know any other work on this approach, for the English language and certainly not for the Hebrew language.

DATA-SET

In this work, we have performed experiments on ANNs whose purpose is to learn to identify a subject in a sentence. The data set on which we did the experiments came from SVLM Hebrew Wikipedia Corpus (SVLM Corpus, 2020). This corpus was used by Silber-Varod et al. (2017) as part of a project of phoneme prevalence testing in the Hebrew language. The sentences originally came from Hebrew Wikipedia. The input of the training and testing set contained 35,000 sentences. We divided the data set into two parts, 75% for the training process and 25% for the testing.

PREPARING THE DATA

For the training step, we tagged the sentences by Dependencies Hebrew Parser (Goldberg, 2011). To test the results of our ANN, we used the same tagger and compared the results of our ANN with Goldberg's tagger results.

PRE-PROCESSING WORD EMBEDDING

We ran Mikolov's word2vec algorithm by applying the gensim tool (Gensim, n.d.) with the following hyper-parameters:

- **min_count** = 1: Minimum words appearance to build for vector, i.e., we build vectors for all the words.
- **window** = 5: For each word, take the five words before and after it.
- **iter** = 100: The number of iterations for the word2vec algorithm in order to build the word vectors.
- **embedding_dim** = 300: The vector size (per word).

PREPARING THE SENTENCES FOR THE NETWORK

To each sentence of the 35,000 sentences, we appended the subject of the sentence. Sentences are constructed from a series of words, with the subject also appearing at the end of each sentence. The input that the network receives is in the form of vectors of numbers. With the word2vec algorithm, we built a vector representation for each word, which means that each statement is a matrix. It follows that the input that the network receives is a representative matrix of the sentence with a special vector representing the subject; this vector appears at the end of the matrix. In the learning phase, the network learns to associate each sentence with its subject, and during the test phase, the network receives a sentence (without it "seeing" its subject) and predicts the sentence subject.

EXPERIMENT

RNNs EXPERIMENTS

We ran the data set mentioned in the previous section on three types of RNN, i.e., LSTM, BiLSTM, and GRU with and without the attention mechanism. The RNNs were written using Keras, which listed in the following URL <https://keras.io>, using Python deep-learning library.

Attention mechanism

The RNN views each word in a sequential order; however, the RNN lacks a view of the entire sentence at once. In order to overcome this drawback, we added the attention mechanism, as we mentioned above. In general, this mechanism works well and improves performance (Bahdanau et al., 2014; Luong et al., 2015). The basic idea of the attention mechanism is similar to that of the human approach. When individuals look for something in a text, they pay attention to certain details concerning their search target, while other details in the sentence are ignored. The same thing is done by attention mechanism: it “sees” the whole sentence and decides which words to give more weight and to which to give less (see Figure 1). In Figure 1, there is input of n -words ($u_{t-n}, \dots, u_{t-1}, u_t$); Bidirectional RNN layer (hidden layers) and attention layer. The words are input to the Bidirectional RNN layer. Each hidden state of the Bidirectional RNN layer is related to a word of the input layer. The attention layer receives the calculated data from the Bidirectional RNN hidden states and then it decides to which words it must “pay more attention”.

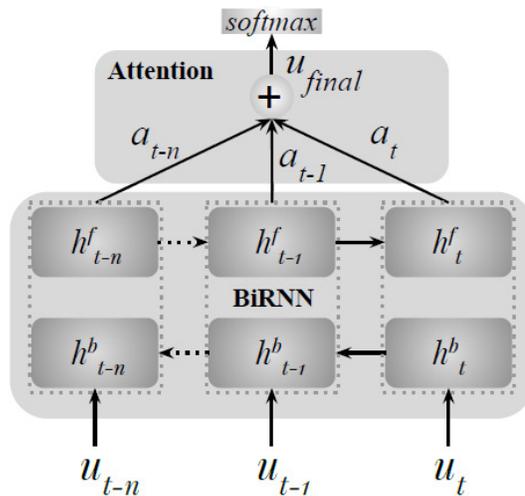


Figure 1. Attention mechanism (Bothe et al., 2018)

In the context of our work, the attention mechanism gives RNNs a look at the whole sentence, which would not be the case without it. This mechanism helps the network decide which words to give greater weight to, as Bahdanau et al. (2014) showed. Figures 2, 3 and 4 show the learning process of the RNNs, i.e., the convergence of LSTM, BiLSTM, and GRU networks, with and without attention mechanism. In Figure 2, we can see the loss-function of the LSTM network with and without attention mechanism; the same is for Figures 3 and 4. The X axis shows the number of iterations of the learning process; the Y axis shows the percentage of network errors.

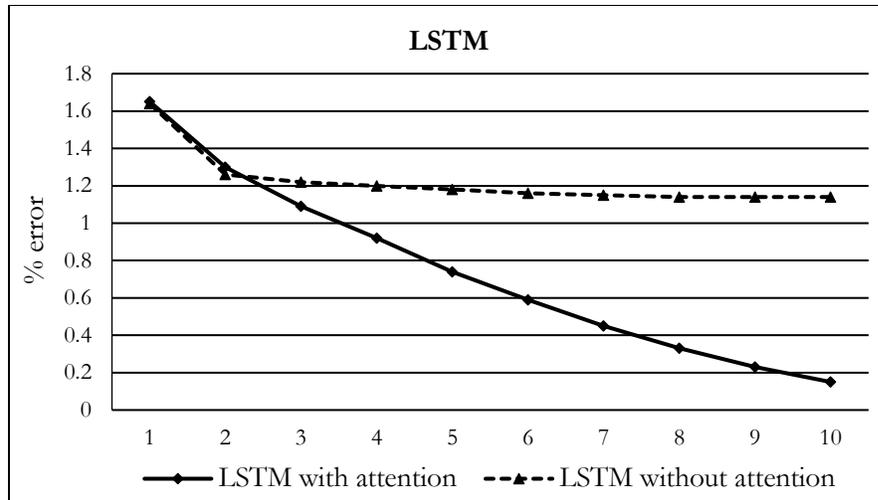


Figure 2. LSTM network with and without attention mechanism

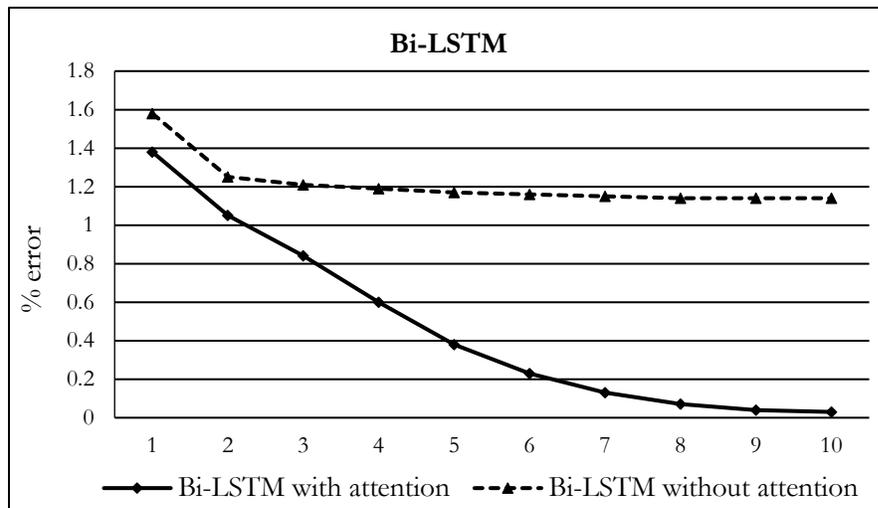


Figure 3. Bi-LSTM network with and without attention mechanism

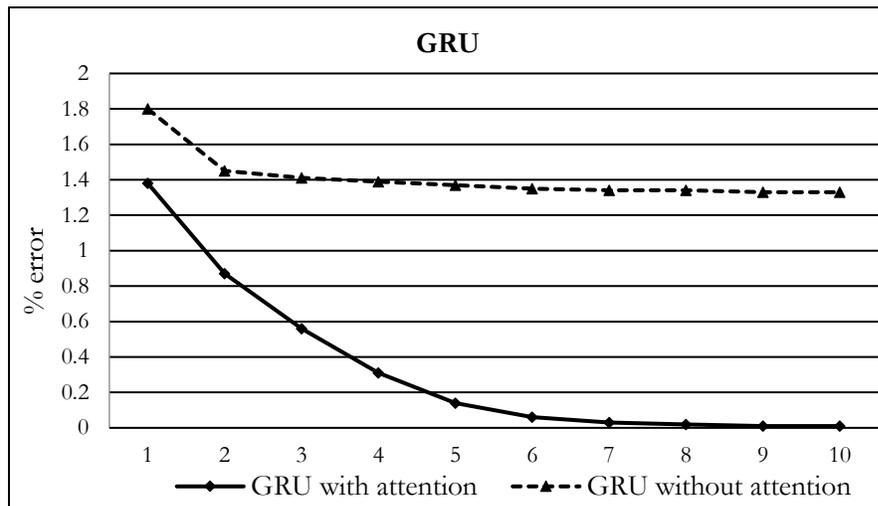


Figure 4. GRU network with and without attention mechanism

We see that in LSTM, BiLSTM, and GRU without attention mechanism there is very similar behavior: a big improvement in the first two iterations, and then small linear improvement up to the tenth iteration.

With the attention mechanism, one sees a beautiful convergence of all three RNNs, unlike without the attention mechanism. The networks with attention mechanism provide better results than without the attention mechanism (as Bahdanau et al., 2014, and Luong et al., 2015, stated) on all the iterations; there is one case where without the mechanism the result is better (the second iteration on the LSTM network). Figures 2, 3 and 4 show important information; the results of the networks with the attention mechanism show both that their results are better, and also that the rate of convergence of learning is much faster than without the attention mechanism.

Here is an example of how a general overview can help. Suppose there is a person who is interested in a particular topic. This person wants to read a paragraph or two about it; however, the text at his disposal is in his second language. If, before he starts reading, he would receive an overview of the subject, he would know what to expect from the text, and then it would be easier for him to read the text. Another example: Suppose there is a newcomer to a language; their first step is to learn the language. During their first period, it takes them a long time to learn the language. While they are learning the language, their mind is busy translating the information they receive; only after that their brain deals with understanding what they read. When they have enough vocabulary, their learning curve grows. Over time, their knowledge has increased, so they have less to learn, and their learning curve is smaller. A similar thing happens in the ANN learning process.

EXAMINING THE SUBJECT-IDENTIFICATION RESULTS

Now we will present randomly selected sentences (which were identified incorrectly) of the output of the ANN. From the examination that we did, we saw that in a significant part of the ANN's errors, the network identifies the central/important word in the sentence as the subject. These words are an important element of the central message of the sentence, but syntactically they are not the subject of the sentence. Another mistake that the ANN makes is the identification of the central noun in a sentence as the subject. This mistake is less common.

The following are examples of sentences that the ANN labeled incorrectly (in Hebrew with English translation). The subjects of sentences appear in **bold**, and the mistakes appear in underline.

- (1) “**ערך** זה עוסק בוועדה לחקר אירועי מלחמת העולם השנייה”
(1) “This **entry** deals with the committee that investigated the events of World War II.”
- (2) “**בית** הפרלמנט נמצא בארמון וסטמינסטר בעיר הבירה לונדון”
(2) “The **seat** of the Parliament is in Westminster Palace in the capital city, London.”
- (3) “**הסברים** שונים לגבי העלאת תמונות תוכן למצוא בויקיפדיה תמונות וקבצים”
(3) “**You** can find various explanations about uploading photos in Wikipedia pictures and files.”
- (4) “**מאידך** ויקישיתוף הוא אוסף הצילומים של קרן ויקימדיה”
(4) “On the other hand, **Wikimedia-Commons** is the Wikimedia Foundation’s photo collection.”
- (5) “**הוא** מזוהה בעיר עם קהילת יהודי סלוניקי.”
(5) “**He** is identified in the city with the Thessaloniki Jewish community.”

SURVEY OF MIDDLE SCHOOL AND HIGH SCHOOL STUDENTS

In this study, we hypothesized that that humans can learn from neuronal network errors and draw practical conclusions for humans. To test our hypothesis, we conducted a survey of 7th graders and of 11th graders. The 7th graders students had not yet learned to perform a syntactic analysis of a

sentence, while the 11th grade students had already taken high school final exams on the syntactic analysis of a sentence.

We distributed a total of 50 sentences to five seventh grade students. The students were asked to label the subject of each sentence. We gave the same 50 sentences to five eleventh grade students, who were also asked to label the subject of each sentence.

We studied all the sentences that the students analyzed. The 7th graders correctly labeled 26% of the sentences, i.e., 13/50. Of the 37 sentences that 7th graders incorrectly labeled, 67.57% were incorrectly labeled in the same way as ANN did. Unlike the seventh graders, the 11th graders correctly labeled 76% of the sentences; that is, 38/50 sentences were correctly labeled. Of the 12 incorrectly labeled sentences, 75% were labeled in the same way as the ANN did.

Common mistakes of the students: (1) Identifying the central message of a sentence as its subject. (2) Identifying an important noun in a sentence as its subject. These two errors are due to the centrality/importance of a word in a sentence. In the human consciousness, a person will “perceive” the central message that the sentence wants to express as its subject. However, syntactically, the central message of the sentence is not necessarily the subject; sometimes the central message of the sentence will be the object of the sentence. The same is true, though less frequently, of a prominent noun. If the sentence revolves around a noun, then there is a likelihood that it will be incorrectly labeled as the subject of the sentence. In both cases, the central issue of the sentence influences the student’s incorrect identification of the subject of the sentence.

PRACTICAL IMPLICATIONS

Practically, at least in our case study, we saw a correlation between mistakes that were made by a machine that mimics a human brain and mistakes that were made by students. So, such machines can be assigned some tasks, and, at the same time, humans can be assigned the same tasks (of course, not every task that humans can perform, a machine can perform, at least not nowadays). While doing these tasks, both machines and humans make mistakes. While recruiting people to do a task can be a long and expensive process, running a process on a computer is a fast and very cheap process. Therefore, we should consider running a computer program as an alternative (or at the very least an aide) to human error surveys and tests. In addition, the computer, unlike humans, does not tire and can easily “answer” thousands of questions.

From the second part of the experiment, it can be seen that a machine that receives an overview of an issue can learn better and faster. From this, it can be tentatively concluded that a similar approach to school, university, or industry instruction will help to achieve a faster, higher quality and cheaper learning process.

This new approach can at least strengthen existing hypotheses and may give quantitative/numerical results (such as the ANN learning process) for problems that are difficult to quantify. It is reasonable to hope that examining ANN results can give new insights, suggest new ideas, and point to new directions, that without the use of ANN it would not be easy to discover.

CONCLUSIONS

In this study, we looked at what humans can infer about themselves from use of RNN (which is a type of ANN). We did the experiment on RNN because RNN is designed to mimic certain actions of the human brain. The experiment we did related to identifying the syntactical subject of Hebrew sentences. We did a survey of middle school students and of high school students who had finished studying Hebrew syntax. The results from the RNN experiment showed that the machine often makes mistakes in finding a syntactic subject in the sentence and incorrectly identifies the central idea of the sentence as its subject. The survey we did of middle school students revealed that these students made mistakes similar to those of the computer. In a survey we did of the high school students,

we found that they also made such mistakes, although at a much lower rate. Thus, the machine process is an inexpensive and efficient way to discover mistakes students are likely to make in the course of the learning process.

We have also shown that when the RNN receives input that contains an overview of the sentence, its learning process improves significantly, both in terms of quality of the results and of the speed of the learning process. We hypothesize that teaching and transferring information to humans similarly, i.e., introducing the topic with a brief overview, will lead to similar improved results of the human teaching process as well.

We conclude that before teaching students a task, it is possible and useful to use RNN as a tool to identify mistakes students are likely to make. The results derived from can help teachers to focus on teaching how to avoid (to the extent possible, even if not completely) common mistakes and other problems students have in mastering the material. Another conclusion we came to is that when a teacher starts teaching a topic, especially in a new field, they should present an overview of the topic.

These are the limitations of our research. (1) We assume that RNN mimics the human brain quite well. The reality is that we have not yet fully achieved this goal, and we doubt that we will do so in the near future. (2) We have presented only our own and specific observation of the experiments and results. In order to know if our conclusions can be generalized to other languages (in case of text analysis) or other tasks, more studies and experiments must be performed. (3) One of the problems with ANN (not specific to our current research) is the inability to know why ANN made its decision, regardless of whether the decision was right or wrong. In recent years, researchers are trying to deal with this problem, but so far without great success.

LOOKING AHEAD (FURTHER RESEARCH)

It seems that RNN (and maybe other ML processes) can give us new ideas about, and new approaches to, teaching, and perhaps to other areas.

We plan to take two groups of students of the same age. The first group will learn a task based on conclusions from the RNN errors. The other group will be a control group; the students in this group will study normally, without reference to the machine learning conclusions. At the end, we will ask both groups to perform the same, and we will examine and analyze the results.

We plan to investigate further mistakes in student learning processes that RNN can help us with.

We performed our research on the Hebrew language, which is a Semitic language. We should also investigate if, for other languages, such as English and German, we reach the same conclusions.

In the current research, we used word2vec vectors; we must investigate if the same results are obtained if we use glove vectors. We also must use word2vec and glove vectors created using a much larger corpus.

We must perform our research on a much larger and more representative data-set (sentences).

As a result of this preliminary study, we plan to consider applying this approach to other areas besides teaching.

REFERENCES

- Artificial neural network. (2020, June 3). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Artificial_neural_network&action=history
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). *Neural machine translation by jointly learning to align and translate*. <https://arxiv.org/abs/1409.0473>
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157-166. <https://doi.org/10.1109/72.279181>

- Bothe, C., Magg, S., Weber, C., & Wermter, S. (2018). *Conversational analysis using utterance-level attention-based bidirectional recurrent neural networks*. <https://arxiv.org/pdf/1805.06242.pdf>
- Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4690-4699). <https://doi.org/10.1109/cvpr.2019.00482>
- Eger, S., Youssef, P., & Gurevych, I. (2019). Is it time to swish? Comparing deep learning activation functions across NLP tasks. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. <https://doi.org/10.18653/v1/d18-1472>
- Firth, J. R. (1957). *A synopsis of linguistic theory, 1930-1955*. The Philological Society.
- Gensim. (n.d.) *models.word2vec – Word2vec embeddings*. <https://radimrehurek.com/gensim/models/word2vec.html>
- Goldberg, Y. (2011). *Automatic syntactic processing of Modern Hebrew*. Ben Gurion University of the Negev.
- Gómez-Rodríguez, C., Alonso-Alonso, I., & Vilares, D. (2019). How important is syntactic parsing accuracy? An empirical evaluation on rule-based sentiment analysis. *Artificial Intelligence Review*, 52(3), 2081-2097. <https://doi.org/10.1007/s10462-017-9584-0>
- HaCohen-Kerner, Y., Beck, H., Yehudai, E., & Mughaz, D. (2006, October). Identifying historical period and ethnic origin of documents using stylistic feature sets. *International Conference on Discovery Science* (pp. 102-113). Springer. https://doi.org/10.1007/11893318_13
- HaCohen-Kerner, Y., Beck, H., Yehudai, E., & Mughaz, D. (2010). Stylistic feature sets as classifiers of documents according to their historical period and ethnic origin. *Applied Artificial Intelligence*, 24(9), 847-862. <https://doi.org/10.1080/08839514.2010.514197>
- HaCohen-Kerner, Y., Beck, H., Yehudai, E., Rosenstein, M., & Mughaz, D. (2010). Cuisine: Classification using stylistic feature sets and/or name-based feature sets. *Journal of the American Society for Information Science and Technology*, 61(8), 1644-1657. <https://doi.org/10.1002/asi.21350>
- HaCohen-Kerner, Y., Kass, A., & Peretz, A. (2004). Baseline methods for automatic disambiguation of abbreviations in Jewish law documents. In J. L. Vicedo, P. Martínez-Barco, R. Muñoz, & M. Saiz Noeda (Eds.), *Advances in Natural Language Processing, EsTAL 2004. Lecture Notes in Computer Science*, 3230, 58-69. Springer. https://doi.org/10.1007/978-3-540-30228-5_6
- HaCohen-Kerner, Y., Kass, A., & Peretz, A. (2010). HAADS: A Hebrew Aramaic abbreviation disambiguation system. *Journal of the American Society for Information Science and Technology*, 61(9), 1923-1932. <https://doi.org/10.1002/asi.21367>
- HaCohen-Kerner, Y., Kass, A., & Peretz, A. (2013). Initialism disambiguation: Man versus machine. *Journal of the American Society for Information Science and Technology*, 64(10), 2133-2148. <https://doi.org/10.1002/asi.22909>
- HaCohen-Kerner, Y., & Mughaz, D. (2010, August). Estimating the birth and death years of authors of undated documents using undated citations. In *International Conference on Natural Language Processing* (pp. 138-149). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-14770-8_17
- HaCohen-Kerner, Y., Mughaz, D., Beck, H., & Yehudai, E. (2008). Words as classifiers of documents according to their historical period and the ethnic origin of their authors. *Cybernetics and Systems: An International Journal*, 39(3), 213-228. <https://doi.org/10.1080/01969720801944299>
- HaCohen-Kerner, Y., Schweitzer, N., & Mughaz, D. (2011). Automatically identifying citations in Hebrew-Aramaic documents. *Cybernetics and Systems: An International Journal*, 42(3), 180-197. <https://doi.org/10.1080/01969722.2011.567893>
- Harris, Z. (1954). Distributional structure. *Word*, 10(23), 146-162.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Jaf, S., & Calder, C. (2019). Deep learning for natural language parsing. *IEEE Access*, 7, 131363-131373. <https://doi.org/10.1109/access.2019.2939687>

- Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4, 237-285. <https://doi.org/10.1613/jair.301>
- Koppel, M., Mughaz, D., & Akiva, N. (2003). CHAT: A system for stylistic classification of Hebrew-Aramaic texts. In *Proceedings of the 3rd Workshop on Operational Text Classification Systems (OTC'03)*. Washington, DC.
- Koppel, M., Mughaz, D., & Akiva, N. (2006). New methods for attribution of rabbinic literature. *Hebrew Linguistics: A Journal for Hebrew Descriptive, Computational and Applied Linguistics*, 57, 5-18.
- Koppel, M., Schler, J., & Mughaz, D. (2004). Text categorization for authorship verification. In *Eighth International Symposium on Artificial Intelligence and Mathematics*. Fort Lauderdale, Florida.
- Kulkarni, A., & Shivananda, A. (2019). Deep learning for NLP. In A. Kulkarni & A. Shivananda (Eds.), *Natural language processing recipes* (pp. 185-227). Apress. https://doi.org/10.1007/978-1-4842-4267-4_6
- Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., & Quillen, D. (2018). Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 37(4-5), 421-436. <https://doi.org/10.1177/0278364917710318>
- Liebeskind, C. (2019). Emoji identification and prediction in Hebrew political corpus. *Issues in Informing Science & Information Technology*, 16, 343-359. <https://doi.org/10.28945/4372>
- Liebeskind, C., & Liebeskind, S. (2019). Emoji prediction for Hebrew political domain. In *Companion Proceedings of the 2019 World Wide Web Conference* (pp. 468-477). <https://doi.org/10.1145/3308560.3316548>
- Liebeskind, S., & Liebeskind, C. (2020). Deep learning for period classification of historical Hebrew texts. *Journal of Data Mining & Digital Humanities*, 2020. <https://jdmhdh.episciences.org/6525>
- Liebeskind, C., Nahon, K., HaCohen-Kerner, Y., & Manor, Y. (2017). Comparing sentiment analysis models to classify attitudes of political comments on Facebook (November 2016). *Polibits*, 55, 17-23.
- Luong, M. T., Pham, H., & Manning, C. D. (2015). *Effective approaches to attention-based neural machine translation*. <https://arxiv.org/abs/1508.04025>
- Luque, C., Luna, J. M., Luque, M., & Ventura, S. (2019). An advanced review on text mining in medicine. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(3), e1302 http://www.uco.es/grupos/kdis/wp-content/uploads/draft_advanced-review-Text-Mining-in-Medicine.pdf
- McDonald, S., & Ramscar, M. (2001). Testing the distributional hypothesis: The influence of context on judgments of semantic similarity. In *Proceedings of the Annual Meeting of the Cognitive Science Society* (Vol. 23, No. 23).
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. <https://arxiv.org/abs/1301.3781>
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS'13: Proceedings of the 26th International Conference on Neural Information Processing Systems* (pp. 3111-3119).
- Mozer, M. C. (1988). *A focused back-propagation algorithm for temporal pattern recognition* (Technical Report). Toronto: University of Toronto, Departments of Psychology and Computer Science.
- Mughaz, D. (2003). *Classification of Hebrew texts according to style* [Master's thesis (in Hebrew)]. Bar-Ilan University, Ramat-Gan, Israel.
- Mughaz, D., Fuchs, T., & Bouhnik, D. (2018). Automatic opinion extraction from short Hebrew texts using machine learning techniques. *Computación y Sistemas*, 22(4). <https://doi.org/10.13053/cys-22-4-3071>
- Mughaz, D., HaCohen-Kerner, Y., & Gabbay, D. (2014a) Identifying birth and death years of authors of undated documents using citations and various constraints [Poster presentation]. *Israeli Seminar on Computational Linguistics*. http://cl.haifa.ac.il/iscol14/Posters/Identifying_Birth_and_Death_Years_of_Authors_of_Undated_Documents.pdf
- Mughaz, D., HaCohen-Kerner, Y., & Gabbay, D. (2014b, November). When text authors lived using undated citations. In *Information Retrieval Facility Conference* (pp. 82-95). Springer. https://doi.org/10.1007/978-3-319-12979-2_8

- Mughaz, D., HaCohen-Kerner, Y., & Gabbay, D. (2015, September). Key-phrases as means to estimate birth and death years of Jewish text authors. In *Semantic Keyword-based Search on Structured Data Sources* (pp. 108-126). Springer, Cham. https://doi.org/10.1007/978-3-319-27932-9_10
- Mughaz, D., HaCohen-Kerner, Y., & Gabbay, D. (2017). Mining and using key-words and key-phrases to identify the era of an anonymous text. In *Transactions on Computational Collective Intelligence XXVI* (pp. 119-143). Springer, Cham. https://doi.org/10.1007/978-3-319-59268-8_6
- Mughaz, D., HaCohen-Kerner, Y., & Gabbay, D. (2019a, June). Extracting and tagging unstructured citation of a Hebrew religious document. In *InSITE 2019: Informing Science+ IT Education Conferences: Jerusalem* (pp. 461-473). <https://doi.org/10.28945/4345>
- Mughaz, D., HaCohen-Kerner, Y., & Gabbay, D. (2019b). Text mining for evaluating authors' birth and death years. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(1), Article 7. <https://doi.org/10.1145/3281631>
- One hot. (2020, June 19). In *Wikipedia*. <https://en.wikipedia.org/w/index.php?title=One-hot&oldid=963335901>
- Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532-1543). <https://doi.org/10.3115/v1/d14-1162>
- Ramos, J. (2003, December). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning* (Vol. 242, pp. 133-142).
- Sahlgren, M. (2008). The distributional hypothesis. *Italian Journal of Disability Studies*, 20, 33-53.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85-117. <https://doi.org/10.1016/j.neunet.2014.09.003>
- Silber-Varod, V., Latin, M., & Moyal, A. (2017). שכיחות פונמות ורצפי פונמות בעברית בתהליך מונחה-נתונים (The Frequency of Hebrew Phonemes and Phoneme Clusters' Frequency in a Data-Driven Approach.) *Hebrew Journal of Literacy and Language*, 6, 22-36.
- Supervised learning. (2020, June 20). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Supervised_learning&oldid=962249039
- Sutskever, I., Martens, J., & Hinton, G. E. (2011). Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (pp. 1017-1024).
- SVLM Corpus. (2020, June 19). In *github*. <https://github.com/NLPH/SVLM-Hebrew-Wikipedia-Corpus>
- Unsupervised learning. (2020, June 18). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Unsupervised_learning&oldid=963135292
- Vemula, A., Mueller, K., & Oh, J. (2018, May). Social attention: Modeling attention in human crowds. In *2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1-7). IEEE. <https://doi.org/10.1109/icra.2018.8460504>
- Wintner, S. (2004). Hebrew computational linguistics: Past and future. *Artificial Intelligence Review*, 21(2), 113-138. <https://doi.org/10.1023/b:aire.0000020865.73561.bc>
- Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 13(3), 55-75. <https://doi.org/10.1109/mci.2018.2840738>
- Zangeneh, E., Rahmati, M., & Mohsenzadeh, Y. (2020). Low resolution face recognition using a two-branch deep convolutional neural network architecture. *Expert Systems with Applications*, 139, 112854. <https://doi.org/10.1016/j.eswa.2019.112854>

BIOGRAPHIES



Dror Mughaz is a doctoral student in the Department of Computer Science at Bar-Ilan University (BIU) under the supervision of Professor Yaakov HaCohen-Kerner and Professor Dov Gabbay. Dror is also a lecturer at the Computer Science Department in the Jerusalem College of Technology (JCT). Dror is a co-author of 20 papers. Dror's current main research is in Text Mining of temporal issues. Other research fields that he works on are text clustering, citation extraction and analysis, word embedding, n-gram embedding, opinion mining, features and key-phrase extraction, data enrichment and author verification. Much of his research was done on Hebrew texts with a focus on rabbinical texts. He was a member of COST research actions of the EU.



Michael Cohen is a B.Sc student in the Department of Computer Science and M.Ba student in the Department of Business Administration & Data Science at Jerusalem College of Technology. Michael's main research is in NLP By dedicating relations in Hebrew sentences with Keras & Tensorflow. Other research fields that he works on are text analysis, Big-Data solutions for data scientists, scalable data flows, data routing, transformation, and system mediation logic. Among other things Michael is a Backend Big Data Developer.



Sagit Mejahez holds an M.A. degree in Education Systems Administration from the Or Yehuda Center for Academic Studies. She holds a B.A. degree from the Department of Jewish art and from the Department of Land of Israel Studies and Archaeology she also holds a teaching certificate from Bar Ilan University. Sagit is an educator and teacher in the Ministry of Education of the State of Israel she. She has over 20 years of experience in different teaching & management positions in the elementary, middle and high schools. During her career Sagit put a special emphasis on integrating new technologies in the teaching processes. Today, her main job is elementary school education and teaching. As part of her work, she managed a young division in a heterogeneous population school. She taught in middle schools and she worked at a high school for high-risk youth in order to keep them in the community. Today Sagit is a second grade educator.



Tal Ades is a graduated B.Sc in the Department of Computer Science and M.Ba student in the Department of Business Administration & Data Science at Jerusalem College of Technology. Tal's main reaserch is in NLP By dedicating relations in Hebrew sentences with Keras & Tensor-flow. Other fields of interest which he works on are fullstack web programming. Among these things Tal is a java real time backend developer.



Prof. Dan Bouhnik is the head of the Engineering and Computer Science Department in the Jerusalem College of Technology (JCT) and a faculty member in the Information Science Department in Bar Ilan University (BIU) in Israel. He is the author of a number of books used for teaching advanced computer sciences and his professional interests include knowledge management, information Security, Big Data and the infusion of new technologies in the learning environment.