Convolutional Neural Network for Automatic Detection of Sociomoral Reasoning Level

Ange Tato
Université du Québec à Montréal

Roger Nkambou
Université du Québec à Montréal

Aude Dufresne
Université de Montréal

Miriam H. Beauchamp
Université de Montréal

ABSTRACT

We propose a model that employs convolutional neural networks (CNN) to evaluate sociomoral reasoning maturity, a key social ability, necessary for adaptive social functioning. Our model is used in a serious game to evaluate learners. It uses pre-annotated textual data (verbatims) and a coding scheme (SoMoral) applied by experts in psychology. State of the art text classification algorithms (Support Vector Machine, Naïve Bayes, etc.) achieved low results in our context in contrary to the CNN that achieved best results with little fine tuning on the input data representation. We use a simple but efficient input data vectors representation learnt directly from the dataset without loosing the sentences 'semantic'. We present a series of experiments with 5 baseline text classification algorithms and 4 baseline data representation. The results show that our model can predict the level of sociomoral reasoning with about 92% of accuracy. Our findings allow not only to advance the text-mining field but also the user modeling in highly social adaptive systems.

Keywords: Convolutional neural networks, data vectors representation, text classification, moral reasoning, social skills, serious game, learner model.

1. INTRODUCTION

Sociomoral reasoning (SMR) is a socio-cognitive construct essential for appropriate decision-making in social contexts, as well as for social adaptation. It is commonly defined as how individuals think about moral emotions and conventions that govern social interactions in their everyday lives [2]. The ability to predict and identify individual's sociomoral reasoning maturity level is a key step to quantifying peoples' social functioning and can be used to identify those at-risk for maladaptive social behaviour and orient them towards appropriate services. We propose a model and a simple input data representation for predicting the level of SMR maturity of an individual based on the justifications they provide when solving sociomoral dilemmas. A computerized test was designed, the Socio-Moral Reasoning Aptitude Level (So-Moral), in which children and adolescents are presented with visual social dilemmas from everyday life and asked to determine how they would react and provide a justification for their answer [21]. A serious game was designed based on the original tasks, and our model was designed to evaluate subjects using existing verbatims and scoring by experts that use the moral maturity coding scheme inspired by a cognitive-developmental approach [7]. The proposed model can be seen as a supervised text classification task.

Text classification is the task of automatically assigning classes to sentences or documents. There exist several supervised classification algorithms that have achieved good results in text classification tasks (Sentiment analysis [15, 19], topic mining [5], etc.) such as Support Vector Machine (SVM), Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA) or MLP (Multilayer perceptron) [1]. While their primary use has been in image classification and speech recognition, deep learning techniques (such as Convolutional Neural Networks) have recently been used for text classification and have achieved remarkable results [8, 11, 23]. A text document is characterized by the words it contains, and consequently the representation of textual data is only based on its words [10]. Thus, an important feature in text classification is the word vector representation of input data. Bag-of-words (BoW) vectors representation is the simplest and most widely used representation where vectors indicate which words appear in the documents without preserving word order. Vectors from BoW lack semantics and are usually huge and sparse. Alternative solutions have been proposed such as n-gram models [18] (bi-gram, tri-gram, etc.), word2vec or wordnet. However, to be effective, models that use n-gram or word2vec require a huge dataset and sentences or words that are frequently observed. Similarly, the use of wordnet is language dependant.

To benefit from word order and the annotated dataset, we built our classifier using CNN and a simple but effective data representation approach called *class-based representation (CBR)*. CNNs are neural networks with layers representing convolving filters applied to local features [12]. The application of CNN on text classification makes use of the 1D structure (word order) of text data so that each unit in the convolution layer responds to a small region of a document (a sequence or pattern of words) [8]. CNN can extract deep features from data which can improve discriminate classes.

1.1 The *Les Dilemmes* serious game

One of the objectives underlying the development of the proposed CNN is to implement the automated scoring mechanism in a serious video game called *Les Dilemmes*. It is a first-person serious game which aims to assess and train the social reasoning skills of the player. It is a virtual environment offering an interactive context which is emotionally, socially and cognitively rich. Players face different socio-moral dilemmas in a 3D environment in which they have to make decisions and are asked to provide oral justifications for the choices they make. They can also ask the opinions of virtual friends (non-player characters) in the game. Their answers are selected from previous recorded verbatims from the different moral maturity levels according to the coding scheme (SoMoral [2]). The learner (player) model implemented in the learning environment includes 3 keys dimensions: the affective state, the cognitive profile and, the sociomoral reasoning profile. Therefore, sociomoral reasoning skill is part of the player model implemented in the game. As stated in [3], a learner model that can accurately represent the learner longitudinally in a game leads to efficient adaptation, which in turn helps increase player satisfaction and his motivation. To this end, it is important to ensure the effectiveness of the learner model before deploying the system for real uses.

Through this work, we aim to build an effective model of the sociomoral facet of the player. The level of sociomoral reasoning of an individual is determined from its verbal justifications provided when solving the dilemmas. This involves the implementation of a model for automatic measurement of this level during the game. We have a dataset of verbatims coming from the SoMoral experimentation already annotated by experts and a description (a paragraph with key concepts) associated with each different level (or class) of maturity. This paper aims to propose a machine learning model that can accurately assess the sociomoral reasoning skill level of a player based on his verbatim. In our knowledge, there is no research that deals with the automatic classification of sociomoral reasoning skills as part of learner-player social behaviour in serious games.

1.2 Sociomoral reasoning skill levels

The original So-Moral task includes five different levels of sociomoral reasoning [2]: (1) Authoritarian-based consequences, (2) Egocentric exchanges, (3) Interpersonal Focus, (4) Societal Regulation and (5) Societal Evaluation. Transition levels (i.e. 1.5, 2.5, 3.5, 4.5) are used to account for verbatims that provide elements of two reasoning stages and show a sequential progression from one stage to another. Occasionally, a verbatim is assigned to two different closed levels (1 being the maximum deviation) when two independent experts annotate the data for rater reliability purposes.

1.3 Dataset

The dataset consists of a benchmark of 691 verbatims (in French) manually coded by experts. Verbatims are short or long text fragments containing at least one sentence. They are not equally distributed between levels. Table 1 shows the repartition of data where for example levels 4.5 and 5 have a smaller number of verbatims than other levels. Level 5 constitutes the highest level of maturity and it is therefore more rarely attributed to children and adolescent's socio-moral justifications. This implies that certain levels have very few examples to learn from. Of the 691 verbatims, 53 were classified as 0, which means that the verbatim does not represent one of the sociomoral reasoning levels (e.g., the answer provided by the participant was tangential and did not contain a justification of their social response). We do not consider these cases in our study, which reduces our corpus to 638 verbatims.

Table 1. Distribution of verbatims between levels

Class	Freq.	%	Class	Freq.	%
1	232	36.36	**1.5**	11	1.8
2	76	11.92	**2.5**	29	4.6
3	207	32.44	**3.5**	31	4.86
4	40	6.3	**4.5**	3	0.48
5	9	1.5			

2. BASELINE METHODS FOR SENTENCE CLASSIFICATION

Since verbatims are annotated text data, we investigated the use of some existing sentence classification algorithms. In this section, we expose state of the art methods for text classification that have shown good results on similar problems.

2.1 Input representation

Here, we present some representation techniques that we experimented on for determining sociomoral reasoning level.

Bag-of-words (BoW): BoW is a binary word presence representation (indicating whether a word is present or not in a sentence). Each distinct word in the dataset corresponds to a feature in the representation. Each labeled verbatim in the dataset is transformed to a vector of N columns, where N (the vocabulary size) is the total number of distinct words in the entire corpus.

Matrix Tf-idf: Tf-idf (Term Frequency-Inverse Document Frequency) representation allows evaluation of the importance of a term contained in a document relative to a collection. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps adjust for the fact that some words appear more frequently in general.

Dictionary of synonyms: We developed a tool to compare our representation model with an approach similar to that of wordnet and to make use of the concept lists from each level provided by experts). The tool takes two words, and for each word, extracts a set of synonyms from a free access online French synonyms database and then computes the intersection of those sets to determine whether the two words are related or not. The So-Moral scoring manuel provides a description of what types of justifications should be included at each level and a list of concepts that describe each level. This information was used by extracting keywords (we removed stop words). After this process, we obtained a list of 53 words representing all the levels, which are used as a vocabulary set for the data. Each word is represented by a vector of size 1*53. For each word from a verbatim and for each word from the vocabulary list, if the intersection of vectors is not null, then it is given a code of 1, otherwise, it is coded 0.

Word2vec: It is common in sentence classification to use publicly available word2vec vectors that are trained on over 100 billion words from Google news [11]. This technique usually works with sentences in English. Instead of directly using those pretrained vector representations, others try to learn those vectors directly from their dataset. We also attempted to represent our data with word2vec vectors that were trained on our corpus.

2.2 Supervised classification algorithms

There exist several supervised classification algorithms. Among them, we selected ones that generally produce excellent results in text classification.

SVM (Support Vector Machine): The learning algorithm consists of finding a hyperplane, which separates the levels appropriately by limiting the error rate of classification in the new data. The aim is to maximize the distance of the vectors close to the hyperplane for each of the levels, which avoids overfitting. Although this algorithm is more suited to binary class problems, the aim was to explore its behavior on our dataset since it generally provides good outcome on text classification [1, 6].

NB (Naïve bayes): NB is a probabilistic classifier based on the Bayes theorem with a naive assumption of attribute independence [17]. It is generally used in the detection of spam, sentiments analysis and in the medical field. The principle is to compute the posterior probability of the class for a given document, and the class with the highest posterior probability is then assigned to the document. We chose to experiment with NB because it is fast [16] and easy to implement especially in real-time applications.

LSA (Latent Semantic Analysis): LSA is an algorithm that has been developed specifically for mining textual data. This algorithm allows us to take into account semantics, which very few algorithms offer. It is an interesting technique because it does not consider any information related to language processing (meaning of words, dictionaries etc.). This makes it possible to establish relations between a set of documents and the terms it contains by constructing "concepts" related to documents and terms [13, 20].

LDA: LDA is a machine learning technique that has revolutionized the extraction of latent subjects in texts [4]. It tries to create topic clustering of documents that are similar to each other. Each document is represented as a mixture of topics. We trained the LDA model on our 638 verbatims by setting the number of topics to 5 or 9 (depending on the problem). The classification of a new verbatim was achieved by computing the cosine similarity between the verbatim and each of the topic probability distribution vectors over words.

MLP (Multilayer Perceptron): MLP is a feedforward artificial neural network model with one or more layers between hidden layers that maps sets of input data onto a set of appropriate outputs. MLPs are widely used for pattern classification, recognition, prediction and approximation. MLPs are able to learn non-linear models, but require tuning of a number of hyperparameters such as the number of hidden neurons, layers, and iterations.

3. THE PROPOSED MODEL

3.1 Class-based Representation (CBR)

The poor distribution of verbatims over levels (see Table 1) and the fact that some of the keywords that can aid in the discrimination of levels appear just once or twice in the entire corpus, makes the application of some data representation techniques inaccurate (e.g. BoW, tf-idf). Also, verbatims are sentences that are semantically very rich and of varying sizes; state-of-the-art techniques often fail to accurately classify this type of data (see Section 5 for details).

We propose a simple yet fast and efficient representation model for data that use only an annotated dataset. Using this technique, we gain over 10% accuracy compared to all other classification techniques previously presented, and over 30% accuracy compared to some state-of-the-art representation models. A further advantage of the proposed representation is that it is not language dependent. It does not consider any information related to language processing (meaning of words, dictionaries etc.), which can be time consuming.

We represented each verbatim as a feature vector, whose values (1 or 0) accounted for the presence of a word in a level. The idea of CBR is simple: if a word appears in the verbatim of one level, then it must be semantically correlated with that class. In turn, if a word appears in verbatims from different classes, then it must be semantically correlated with all the classes, but has less significance than a word that appears only in verbatims from one of those levels. For example: we have 4 levels, and we have 2 verbatims from level1 and level 2 (see Table 2a); Table 2b shows the representation of two words in this specific case (1 means semantically correlated and 0 means uncorrelated).

Table 2. a) Examples of two verbatims

Verbatim	Class
*Parce que c'est mal et elle n'apprendrait pas de ses erreurs (*Because it's wrong and she won't learn from her mistakes)	3
C'est tricher (It's cheating)	1

b) Examples of CBR on the 2 verbatims from a).

Words\Classes	1	2	3	4
est	1	0	1	0
erreurs	0	0	1	0

The input data for the CNN model is a matrix with 5 columns and 88 lines, which correspond to the length (number of words) of the longest sentence of the corpus after data pre-processing.

3.2 The CNN Model

According to LeCun and colleagues [14], deep learning allows computational models composed of multiple layers of processing to learn data representations at multiple levels of abstraction. Deep learning techniques such as CNN have been shown to be effective for Natural Language Processing (NLP) and have achieved excellent results on sentence classification [11, 24], sentence modeling, and semantic parsing [9]. They can explore small text regions to learn useful features for categorization [8]. The CNN we are proposing requires as input a vector representation (88*5 or 88*9) of verbatims that preserves the internal order of words, as in class-based representation.

Parameter selection: The hyper-parameters of our CNN, such as the size of filters and the number of layers, were chosen based on the results obtained empirically from several tests on our dataset. The structure of the CNN consists of two layers of *convolution*, two layers of *maxpooling* and one layer fully *connected* to the output. The fully connected layer of our model uses 40 rectified linear units. The structure also includes two Filter windows, one of size 1x5 for the 5-level classifier (1x9 for the 9-level classifier) and the other of 2x1 in size. The first filter window is used to implement the convolution on the input data. Using a 1-dimension window here allows exploration of the data one word at a time in order to derive specific features associated with each word (which contributes to determining the semantics of the word). Following this step and the maxpooling of its output, another filter is used for a second convolution. This second convolution aims at extracting features related to word order (or text regions). A filter vector of 2x1 (for exploring the text regions) is used for this purpose. There are 20 filters in each convolutional layer. The *batchsize* was set to 500 and the number of iterations to 250.

4. EXPERIMENTS

Our experiments involve the five classification algorithms, Naive Bayes (NB), LSA (Latent semantic Analysis), LDA

(Latent Dirichlet Allocation), MLP (Multi Layer Perceptron) and Support Vector Machine (SVM) that we presented earlier in this paper. The goal of using all these algorithms is to compare the models obtained from them with that obtained from the CNN-based model. We explored existing input representations of data and compared results with the CBR.

4.1 Data pre-processing

For consistency between different input representations and algorithms, we used the same pre-processing steps for the data.

Stop-word removal: Generally, the very first step to reduce the vector size of the data is to remove stop-words (connective words, such as “a”, “in”, “the” in English). Alone, they are considered lacking semantic to give information to the classifier [1]. Unfortunately, the typical list of stop words for the French available online gave poor results in our classification task. Instead, we excluded common words in the verbatims, which were not discriminatory for the different SMR levels.

Lemmatization: This is the process of mapping words onto their base form [10]. For example, the words “installed”, “installs” and “installing” are mapped to “install”. This mapping makes the binary presence of word representation approaches treat words of different forms as a single feature, hence reducing the total number of features. We used the Stanford NLP tools to apply a French lemmatization to the verbatims.

4.2 Results

Accuracy is typically used as the standard measure for classification performance. However, for datasets with an unbalanced distribution such as the one used here, this measure can be illusory and not very informative about the errors being committed by the classifier. Instead of relying solely on accuracy, we used the F1 score (or F-measure) which takes into consideration both precision and recall. To provide a point of reference for our CNN model results using our proposed input representations, we first report the performance achieved using baseline techniques for sentence classification. We report Accuracy and F1-score over all datamining techniques and datasets in Tables 3 and 4. First, we used only the BoW and tf-idf representations as input representation for the algorithms. SVM was run with the RBF (Radial Basis Function) as kernel function. LSA is an algorithm which initially works with tf-idf representation, that is why we have the n/a (not applicable) mention. Table 3 shows the results. For a second experiment, we used dictionary of synonyms and the *CBR* representation as input representation techniques for the MLP and the CNN. We have kept only those 2 algorithms for the next step because of their good results compared to others on step 1.

Table 4 shows results. **Erreur ! Source du renvoi introuvable.**Figure 1 graphically shows the performance of MLP and CNN on both the dictionaries synonyms and *classes-based* techniques and on the 2 types of problems previously mentioned in section 2 (5 and 9 classes).

For all the algorithms, we trained the models on 75% of data (which is about 500 verbatims) and we tested on the remaining verbatims (138 verbatims).

5. DISCUSSION

We begin our discussion by looking at the most basic representations, those involving the BoW and the tf-idf (Table 3). We note that none of the 5 baseline algorithms were able to classify at least the half of the data with the BoW representation technique. Only NB and MLP were able to classify more than 50% with tf-idf. However, the F1 score remains relatively low in general. Furthermore, for SVM, LSA and LDA, all the verbatims in the test data were classified as level 1. For NB, they were classified into levels 1 and 3. The reason for these misclassifications can be seen in Table 1, where levels 1 and 3 are the most represented in the dataset. This brings us to the conclusion that those 2 representations depend strongly on the distribution of the data into classes. Despite the time-consuming learning, CNN and MLP gave the best results.

Table 3. Accuracy and F1 scores of 6 baseline algorithms for sociomoral reasoning level classification. The input data are represented with BoW and Tf-idf techniques.

Input representation	Measure	SVM	NB	LSA	LDA	MLP	CNN
BoW	**Accuracy**	43.00	49.28	N/A	38.96	49.91	**49.98**
	F1-score	12.71	**30.55**	N/A	25.24	28.32	29.18
Tf-idf	**Accuracy**	43.27	59.9	31.05	48.36	60.25	**63.00**
	F1-score	18.54	**46.81**	15.4	44.00	45.79	37.68

Table 4. Accuracy and F1 score of the CNN model and MLP for sociomoral reasoning level classification. The input data are represented with class-based and dictionary of synonyms techniques.

Input representation	Measure	MLP (5 classes)	CNN (5 classes)	MLP (9 classes)	CNN (9 classes)
Dictionary of synonyms	**Accuracy**	66.33	**71.25**	44.00	52.00
	F1-score	60.4	54.34	36.21	**64.37**
CBR	**Accuracy**	75.00	**85.8**	56.60	82.60
	F1-score	67.76	**83.76**	37.09	74.8

Table 5. Results of the CNN model and MLP with errors margins (1 for the 5 classes and 0.5 or the 9 classes).

Input representation	MLP (5 levels)	CNN (5 levels)	MLP (9 levels)	CNN (9 levels)
CBR	84.28	**92.00**	63.52	**84.00**

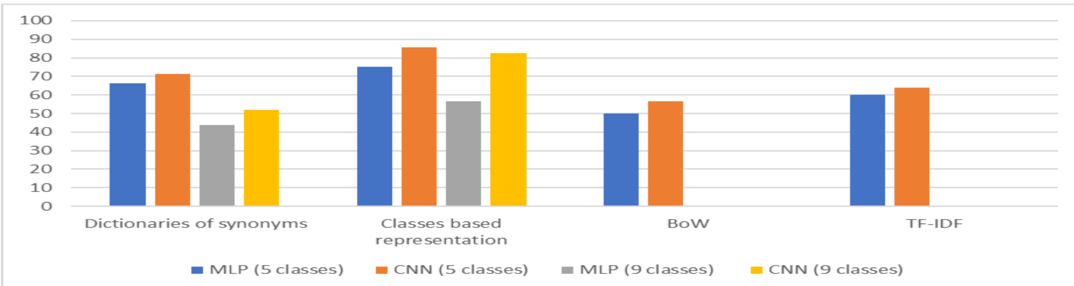

Figure 1. Variation of the accuracy of MLP and CNN, based on input data representation techniques.

In

Table 4, we ran our CNN model and MLP with the dictionary of synonyms and class-based techniques. We also considered the 5 and 9 levels problem. At first glance, we see that the CBR gives the best results compared to other representation techniques. The best result was obtained from our CNN model. The model provided 85% accuracy and 83% F1 score, which is an acceptable result for the problem. The training of the CNN took more time than other techniques because we needed to find the parameters that achieved the best results. We limited the number of iterations to 250 to avoid overfitting. Over the 250 iterations, we obtained poorer classification results on test data, but over 99% on training data. We also note that the results vary considerably based on the training set, suggesting that selection of the training set is an important part in the pre-processing of data for the CNN model.

Why does CNN give the best results?

The size of filters (number of lines) in the CNN can be compared to the idea behind N-Grams. The convolution is done on 2, 3 or 4 words at a time if the filters are respectively of size 2, 3 and 4. So, the CNN takes into account the order of the words in sentences. Another reason for the better results with the CNN compared to other techniques is that it can extract deep features (e.g., semantically grounded) using a series of convolutions, filters, feature maps and pooling on data, which help in the discrimination of data. The input data representation also contributes to this performance.

Real-life sociomoral reasoning classification

In manual scoring of socio-moral reasoning, different experts occasionally associate the same verbatim with different levels because of inherent variability between even expert raters. Taking into consideration that even experts can make errors, we retrained our model (on both 5-level and 9-level problems) by considering a margin error of 1 for the 5-level problem and of 0.5 for the 9-level problem. For example, if the model predicts that the level of verbatim v1 is 1 and that the real level is 1.5, then it is considered as a true classification.

Table 5 shows the results when error margins are considered. We can see that the CNN on the 5-level problem achieves exceptional results with an accuracy of 92%, which is the best so far.

6. CONCLUSION

We propose a model able to predict with over 90% accuracy the sociomoral reasoning skill level based on a textual verbatim. Specifically, we propose a simple but efficient input text data representation that can work with different classification algorithms. This work is a considerable contribution in sentence classification and in sociomoral reasoning maturity classification. Verbatims are typically manually annotated by experts. Our proposed model is intended to help them in this task and produces results that are comparable to the accuracy of independent raters, suggesting promising applications.

Contrary to state-of-the-art techniques in text classification, the CNN model we propose achieves the best results in our context. This is mainly due to its deep structure that can learn useful features from data. Despite the good results obtained by the CNN, parameters must be manually tuned and require many experiments to find the best results. MLP can be treated as a lexical mining technique on text, because all neurons on hidden layers receive information from all previous neurons (blind mining). The order or the meaning of words is not considered. On the other hand, CNN can capture deep features from data and thus the order (pattern or syntax mining) and the meaning (semantic mining) of words, if the representation is good enough. Since a sentence is fully defined by its syntax, lexis and semantics, a model considering those features will lead to better results in sentence classification and even NLP tasks. In our future works, we will develop a model based on a pooling of MLP and CNN techniques. We will also consider the use of the multiple channels features of CNN to combine different representation of sentences as reported by Kim [11] and Yin [22]. Similarly, while more complex data representations for text classification will undoubtedly continue to be developed, those deploying such technologies in real-life problems will likely be attracted to simpler variants, which afford fast training and prediction times such as the CBR model that we propose. The only downside of our representation approach is that it requires a classified dataset. We will explore the combination of class-based approach and others interesting representation techniques that use RBM (Restricted Boltzmann Machine) or autoencoders in future work, in order to achieve 90% accuracy without adjustment for error margins. The proposed coding solution will be implemented in *the Les Dilemmes video game*. The next step will be the assessment of the efficiency of the sociomoral reasoning dimension as a learner model facet in a highly adaptive social serious video game.

7. ACKNOWLEDGMENTS

We would like to thank our colleagues from ABCs research lab who provided us with the So-Moral data.

8. REFERENCES

[1] Aggarwal, C.C. and C. Zhai, *A survey of text classification algorithms*, in *Mining text data*. 2012, Springer. p. 163-222.

[2] Beauchamp, M., J.J. Dooley, and V. Anderson, *A preliminary investigation of moral reasoning and empathy after traumatic brain injury in adolescents.* Brain injury, 2013. **27**(7-8): p. 896-902.

[3] Birk, M.V., et al. *Modeling Motivation in a Social Network Game using Player-Centric Traits and Personality Traits*. in *International Conference on User Modeling, Adaptation, and Personalization*. 2015. Springer. p. 18-30.

[4] Blei, D.M., A.Y. Ng, and M.I. Jordan, *Latent dirichlet allocation.* Journal of machine Learning research, 2003. **3**(Jan): p. 993-1022.

[5] Chen, Z. and B. Liu. *Mining topics in documents: standing on the shoulders of big data*. in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014. ACM. p. 1116-1125.

[6] Gautam, G. and D. Yadav. *Sentiment analysis of twitter data using machine learning approaches and semantic analysis*. in *Contemporary computing (IC3), 2014 seventh international conference on*. 2014. IEEE. p. 437-442.

[7] Gibbs, J.C., *Moral development and reality: Beyond the theories of Kohlberg, Hoffman, and Haidt*. 2013: Oxford University Press.

[8] Johnson, R. and T. Zhang, *Effective use of word order for text categorization with convolutional neural networks.* arXiv preprint arXiv:1412.1058, 2014.

[9] Kalchbrenner, N., E. Grefenstette, and P. Blunsom, *A convolutional neural network for modelling sentences.* arXiv preprint arXiv:1404.2188, 2014.

[10] Khoo, A., Y. Marom, and D. Albrecht. *Experiments with sentence classification*. in *Proceedings of the 2006 Australasian language technology workshop*. 2006. p. 18-25.

[11] Kim, Y., *Convolutional neural networks for sentence classification.* arXiv preprint arXiv:1408.5882, 2014.

[12] Krizhevsky, A., I. Sutskever, and G.E. Hinton. *Imagenet classification with deep convolutional neural networks*. in *Advances in neural information processing systems*. 2012. p. 1097-1105.

[13] Landauer, T.K., *Latent semantic analysis*. 2006: Wiley Online Library.

[14] LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* Nature, 2015. **521**(7553): p. 436-444.

[15] Maas, A.L., et al. *Learning word vectors for sentiment analysis*. in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. 2011. Association for Computational Linguistics. p. 142-150.

[16] Narayanan, V., I. Arora, and A. Bhatia. *Fast and accurate sentiment classification using an enhanced Naive Bayes model*. in *International Conference on Intelligent Data Engineering and Automated Learning*. 2013. Springer. p. 194-201.

[17] Rish, I. *An empirical study of the naive Bayes classifier*. in *IJCAI 2001 workshop on empirical methods in artificial intelligence*. 2001. IBM New York. p. 41-46.

[18] Sidorov, G., et al., *Syntactic n-grams as machine learning features for natural language processing.* Expert Systems with Applications, 2014. **41**(3): p. 853-860.

[19] Tang, D., et al. *Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification*. in *ACL (1)*. 2014. p. 1555-1565.

[20] Tougas, J.E. and R.J. Spiteri, *Updating the partial singular value decomposition in latent semantic indexing.* Computational Statistics & Data Analysis, 2007. **52**(1): p. 174-183.

[21] Vera-Estay, E., et al., *All for One: Contributions of Age, Socioeconomic Factors, Executive Functioning, and Social Cognition to Moral Reasoning in Childhood.* Frontiers in psychology, 2016. **7**.

[22] Yin, W. and H. Schütze. *Multichannel variable-size convolution for sentence classification*. in *Proceedings of the Conference on Computational Natural Language Learning*. 2015. p. 204-214.

[23] Zhang, X. and Y. LeCun, *Text understanding from scratch.* arXiv preprint arXiv:1502.01710, 2015.

[24] Zhang, Y. and B. Wallace, *A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification.* arXiv preprint arXiv:1510.03820, 2015.