Optional ERIC Coversheet — Only for Use with U.S. Department of Education Grantee Submissions

This coversheet should be completed by grantees and added to the PDF of your submission if the information required in this form **is not included on the PDF to be submitted**.

INSTRUCTIONS

- Before beginning submission process, download this PDF coversheet if you will need to provide information not on the PDF.
- Fill in all fields—information in this form **must match** the information on the submitted PDF and add missing information.
- Attach completed coversheet to the PDF you will upload to ERIC [use Adobe Acrobat or other program to combine PDF files]—do not upload the coversheet as a separate document.
- Begin completing submission form at https://eric.ed.gov/submit/ and upload the full-text PDF with attached coversheet when indicated. Your full-text PDF will display in ERIC after the 12-month embargo period.

GRANTEE SUBMISSION REQUIRED FIELDS

Title of article, paper, or other content

The Automated Model of Comprehension Version 3.0: Paying Attention to Context

All author name(s) and affiliations on PDF. If more than 6 names, ERIC will complete the list from the submitted PDF.

Last Name, First Name	Academic/Organizational Affiliation	ORCID ID
Corlatescu, Dragos	University Politehnica of Bucharest, Bucharest	
Watanabe, Micah	Department of Psychology, Arizona State Univ	
Ruseti, Stefan	University Politehnica of Bucharest, Bucharest	
Dascalu, Mihai	University Politehnica of Bucharest, Bucharest	
McNamara, Danielle S.	Department of Psychology, Arizona State Univ	

Publication/Completion Date—(if *In Press,* enter year accepted or completed) 26 June 2023

Check type of content being submitted and complete one of the following in the box below:

- ☐ If article: Name of journal, volume, and issue number if available
- ☑ If paper: Name of conference, date of conference, and place of conference
- ☐ If book chapter: Title of book, page range, publisher name and location
- ☐ If book: Publisher name and location
- ☐ If dissertation: Name of institution, type of degree, and department granting degree

AIED 2023: Artificial Intelligence in Education, July 3-7 2023, Tokyo, Japan

DOI or URL to published work (if available) https://doi.org/10.1007/978-3-031-36272-9_19

Acknowledgement of Funding— Grantees should check with their grant officer for the preferred wording to acknowledge funding. If the grant officer does not have a preference, grantees can use this suggested wording (adjust wording if multiple grants are to be acknowledged). Fill in Department of Education funding office, grant number, and name of grant recipient institution or organization.

"This work was supported by U.S. Department of Education **[Office name]** National Center for Educat through **[Grant number]** R305A1900 to **Institution]** Arizona State U.The opinions expressed are those of the authors and do not represent views of the **[Office name]** National Center for Educa or the U.S. Department of Education.

The Automated Model of Comprehension Version 3.0: Paying Attention to Context

Dragos Corlatescu[1,2], Micah Watanabe[3], Stefan Ruseti[1], Mihai Dascalu[1,4(✉)], and Danielle S. McNamara[3]

[1] University Politehnica of Bucharest, 313 Splaiul Independentei, 060042 Bucharest, Romania
{dragos.corlatescu,stefan.ruseti,mihai.dascalu}@upb.ro
[2] CrowdStrike, 201 Str. Barbu Văcărescu, 020285 Bucharest, Romania
dragos.corlatescu@crowdstrike.com
[3] Department of Psychology, Arizona State University, PO Box 871104, Tempe, AZ 85287, USA
{micah.watanabe,dsmcnama}@asu.edu
[4] Academy of Romanian Scientists, Str. Ilfov, Nr. 3, 050044 Bucharest, Romania

Abstract. Reading comprehension is essential for both knowledge acquisition and memory reinforcement. Automated modeling of the comprehension process provides insights into the efficacy of specific texts as learning tools. This paper introduces an improved version of the Automated Model of Comprehension, version 3.0 (AMoC v3.0). AMoC v3.0 is based on two theoretical models of the comprehension process, namely the Construction-Integration and the Landscape models. In addition to the lessons learned from the previous versions, AMoC v3.0 uses Transformer-based contextualized embeddings to build and update the concept graph as a simulation of reading. Besides taking into account generative language models and presenting a visual walkthrough of how the model works, AMoC v3.0 surpasses the previous version in terms of the Spearman correlations between our activation scores and the values reported in the original Landscape Model for the presented use case. Moreover, features derived from AMoC significantly differentiate between high-low cohesion texts, thus arguing for the model's capabilities to simulate different reading conditions.

Keywords: Reading Comprehension · Automated Model of Comprehension · Language Model

1 Introduction

Reading comprehension is a foundational skill for both knowledge acquisition [9] and memory reinforcement [1]. Successful reading comprehension depends on both individual factors (e.g., decoding ability, goals, motivation, and prior knowledge) and features of the text (e.g., syntactic complexity and cohesion).

N. Wang et al. (Eds.): AIED 2023, LNAI 13916, pp. 229–241, 2023.
https://doi.org/10.1007/978-3-031-36272-9_19

Importantly, these two factors (individual features and text features) have interactive effects. For example, less skilled readers benefit from highly cohesive texts, whereas skilled readers benefit from less cohesive texts [11]. While the theoretical effects of individual and text features on comprehension have been documented, predicting the effects across different texts depends on effective modeling of the comprehension process. Therefore, modeling the comprehension process across numerous texts can provide information on the efficacy of specific texts as learning tools prior to their use in research and classrooms.

The Automated Model of Comprehension (AMoC) provides teachers and researchers the ability to model the reading comprehension process of different texts. AMoC leverages the general frameworks provided by the Construction-Integration (CI) Model and Landscape Model and integrates modern language models and natural language processing tools in order to simulate the readers' mental representation during reading. When a text is analyzed with AMoC, a concept graph is produced, which represents the generation of concepts (nodes) and inferences (edges). While previous iterations of AMoC have been tested and published, the current version uses state-of-the-art language models that are superior to those used in previous versions.

The Automated Model of Comprehension has its roots in the Construction-Integration model [10] and the Landscape Model [1]. The CI model describes comprehension as a two-step process. Readers first *construct* a mental representation of a text's meaning by generating inferences, and then *integrate* the new, text-based information with their own existing knowledge. The Landscape Model simulates the activation of concepts in a similar way to the CI Model; however, it is further used to simulate the fluctuation of activated concepts across time. Within the Landscape Model, prior knowledge can be activated through two different mechanisms: cohort activation, which is the passive linking of concepts related to the reader's mental representation through the formation of associative memory traces, and coherence-based retrieval, which represents the importance of a word in relation to the text (such as causal, temporal, or spatial connections). The coherence parameter is smaller when the reading process is more superficial. Importantly, both the CI model and the Landscape model were automated, but the parameters and connections were entirely hand-sewn - meaning the experimenter listed the nodes and connections, as well as the assumed knowledge activated by the reader, and the models simply'put' the pieces together. Our objective is to automate the entire reading process, including which nodes are activated, their connections, and whatever outside knowledge is activated during the reading process.

Two versions of the Automated Model of Comprehension have been published (i.e., v1.0 [5] and v2.0 [3]), but recent developments in text analysis have enabled improvement in AMoC's internal modeling and output. The third version of AMoC uses the Transformer [16] architecture for taking into account context and having a better representation of words. We publicly released our model and all corresponding code on GitHub[1]. In contrast, AMoC v2.0 [3] employed a

[1] https://github.com/readerbench/AMoC.

combination of word2vec [13] and Wordnet [14] to generate nodes and edges. The main feature of the Transformer is the self-attention mechanism which enables a better understanding of the relations between words in a sentence across a wider span. Figure 1 introduces attention edges generated using BertViz [17] from the "knight" token in the second sentence to its previous occurrence, as well as lexical associations to other words. Additionally, Transformers are faster in both the training phase and inference phase in comparison to Recurrent Neural Networks due to their parallel nature. This architecture has been successfully employed on a variety of NLP tasks for which it holds state-of-the-art performance.

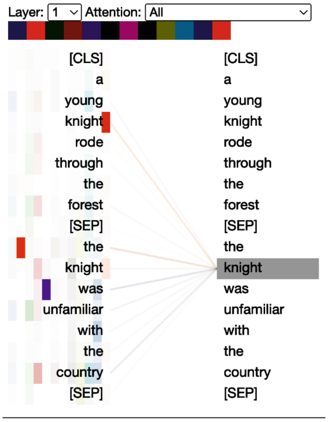


Fig. 1. BertViz self-attentions.

Generative Pre-training Transformer (GPT) is a large Transformer-based generative language model [15]. Instead of focusing on representing word embeddings such as the BERT [6] encoder, GPT's goal is to generate text with cursive ideas that map well with the context and are syntactically correct. As such, the end goal can be considered the generation of text that is indistinguishable from human-generated writing. The training process involved predicting the next word in a sequence given the previous words. Additionally, GPT can be customized to slightly guide its prediction - for example, the length in tokens can be set by the user. N-grams of different lengths can be set not to repeat in order to have a more diverse generation. Sampling can also be used when choosing the next word in a generation to ensure a more diverse output. Multiple GPT models have been released at the time of this writing, namely versions 1–4. GPT-3 and 4 [2] require extensive resources and are available only via API. Since we want to make AMoC available to everyone and deployable on most computers with average specifications, we decided to train and test AMoC v3.0 using GPT-2.

2 Method

AMoC mimics human comprehension by using the generative capabilities of state-of-the-art language models. When analyzing a text, AMoC generates a concept graph. The concept map contains nodes and edges. Each node represents a concept (i.e., a noun, verb, adjective, or adverb) and includes information about the word lemma and whether the concept is text-based or inferred. The edges are weighted links, the weight being derived from the attention scores of the GPT-2 model in different contexts. Here, we consider the mean attention score between 2 words across all layers from GPT-2. The AMOC v3.0 workflow is depicted in Fig. 2 as follows.

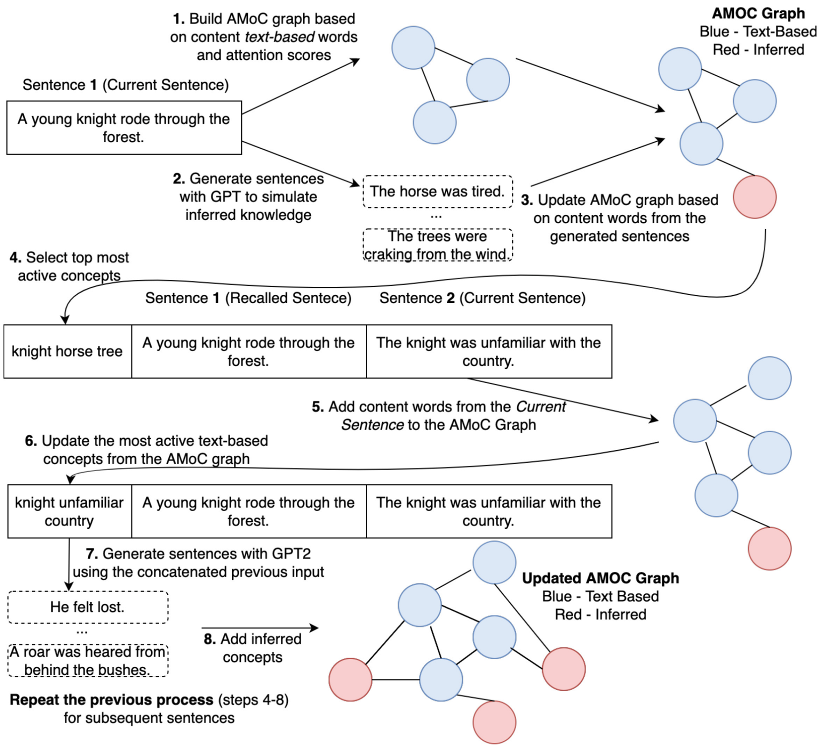


Fig. 2. AMoC v3 processing flow.

2.1 Processing Flow

The text analysis begins with an empty concept graph. There are no previous sentences in the working memory of the model nor any concepts in the AMoC graph. In the first step, the content words from the sentence (i.e., nouns, verbs, adjectives, and adverbs) are added to the graph as nodes with edges connecting them. The weights of the edges are given by the attention scores of the *Model*. In the second step, sentences are generated using the *Model*, while the content words from the generated sentences are added to the graph, along with the edges connecting the generated content words to the text-based content words.

Note that this first sentence scenario is just a sub-scenario of a general analysis presented below.

For each sentence after the first sentence, the model retains active concepts from the *PrevSent* sentences as nodes in the concept graph. In addition, the model retains edges and utilizes a decay factor to mimic reading over time (see below). The *PrevSent* sentences, top *Active* concepts from the concept graph, and the current sentence are given to the model. New nodes are generated, and edges are added based on the attention scores of the content words from the current sentence and the other content words.

New sentences are generated using the *Model* that receives as input the current sentence, past *PrevSent* sentences, and top *Active* nodes from the AMoC graph that are text-based. Note that in the generation process, we do not consider only the concepts that are text-based (i.e. not the inferred concepts). The generation can take multiple forms using the *Imagin* (parameter) to make them more diverse or more clustered on the same idea. We can also vary the number of sentences generated as well as the length of tokens from the generation sequence.

The top *DictExp* number of words that have a maximum attention score greater than a *AttThresh* threshold are added to the graph. This process limits the number of inferred concepts from the generation process as well as imposes a condition that the words should have a strong connection (attention score) with at least one of the words from the "source" text.

When analyzing the next sentence in the text (i.e., simulating the reading of another sentence), the model supports a decay factor (i.e., forgetting the previous sentences) (parameter *WD*) such that all the edges in the graph are decayed with a specified percentage. In addition, the importance of the nodes can be scaled with their Age of Exposure [4] scores (parameter *UseAOE*).

The process repeats with subsequent sentences, past *PrevSent* sentences as a sliding window, top *Active* concepts active from the graph, and corresponding method to update the AMoC graph, generate follow-up sentences and add new inferred concepts to the AMoC graph.

2.2 Features Derived from AMoC

The following list introduces the graph metrics that consider either the entire AMoC graph or only the active nodes from the AMoC graph. In order to obtain the final values, the graph after each sentence was saved, and the metrics were computed on each of the intermediate graphs as the mean value of the targeted nodes. Then, the mean of all of these mean values per sentence was computed for the entire AMoC graph.

- Closeness centrality - a local measure of centrality per node that takes into account the minimum distance from a node to all other nodes;
- Betweenness centrality - a local measure reflecting the number of shortest paths that go through a node;
- Degree centrality - a local measure computed as the sum of the weights to adjacent nodes;

– Density - a global measure of the AMoC graph, which is the ratio between the number of edges and the maximum possible number of edges in the AMoC graph;
– Modularity - a global measure of how difficult it is to split the AMoC graph into multiple subgraphs.

2.3 Experimental Setup

The model considers the following Parameters presented in Table 1. Each of the parameters aligns with assumptions regarding the reader's skill in parsing, understanding, and remembering text, their knowledge of words, and their tendency to go beyond the text using elaboration and imagery. Note that the default values were chosen based on an expert judgment that follows an analogy to the manner in which other comprehension models were evaluated (i.e., the CI and Landscape models).

Table 1. AMoC v3.0 parameters.

Parameter	Description	Default value
Model	Normal (more complex) versus Distilled (less complex, faster) GPT	gpt2
PrevSent	Number of sentences that are remembered in their raw form (max 3)	1
Active	Number of active concepts in the reader's mind	5
DictExp	Maximum dictionary expansion	7
AttThresh	Attention score threshold for new concepts	0.3
Imagin	Values from 1 to 4 denoting potential elaboration and imagery by the reader (1 less imaginative, 4 more imaginative)	1
WD	Percentage of decay of each edge from sentence to sentence	0.1
UseAOE	Binary value indicative of whether to scale by AOE scores when computing the importance of the nodes	1 (True)

2.4 Comparison Between AMoC Versions

The main differences between the three AMoC versions are displayed in Table 2. AMoC v3.0 is implemented in Python like AMoC v2.0 [3], while AMoC v1.0 [5] was in Java. spaCy [8] is also used in both the 2nd and 3rd versions to apply basic processing on the text (i.e., lemmatization, POS tagging for the extraction of content words). The 1st and the 2nd versions of AMoC use word2vec to provide weights for the edges in the concept graph, while in comparison, the 3rd version uses GPT2 attention scores. As mentioned in the introduction section, Transformer models (including GPT2) obtained state-of-the-art results in multiple fields due to the self-attention mechanism, along with other aspects. In addition, the similarity scores obtained by Transformer models are context-related. Therefore, these scores should be of higher quality than other similarity scores that are general and do not take context into account (e.g., scores provided by word2vec). The inference process also differs in AMoC v3.0 compared to the previous versions. AMoC v3.0 uses GPT2 text generation to extract new concepts that are more related to the context of the text, rather than just picking similar words from a language model (word2vec) or WordNet. Lastly, AMoC v3.0 is highly customizable with 8 parameters that can be manually configured to serve the user's needs, surpassing by far the 3 customizable parameters from AMoC v2.0.

Table 2. Comparison between AMoC versions.

Feature	AMoC v1.0	AMoC v2.0	AMoC v3.0
Programming Language	Java	Python	Python
Sentence Segmentation	Stanford Core NLP v3.8	SpaCy v2.3.5	SpaCy v3.4
Tokenization	Stanford Core NLP v3.8	SpaCy v2.3.5	SpaCy v3.4
Language model used for weighting text-based concepts	Word2Vec	Word2Vec	GPT2 attention scores
Inferring new concepts	WordNet + Word2Vec	WordNet + Word2Vec	GPT2 text generation
Adding newly inferred concepts	PageRank	PageRank	GPT2 attention scores
Number of parameters	3	3	8

3 Results

We begin this section by showcasing the stages of the AMoC v3.0 processing flow using real text. Next, we present two experiments conducted on both AMoC v2.0 and AMoC v3.0 to validate the models and also provide a performance comparison between them. The first experiment involves computing correlations between the Landscape Model word scores and the activation scores from AMoC models, while the second experiment analyzes how related are the AMoC graph centrality metrics to the cohesion of the text.

3.1 Use Case

We used the text from the original Landscape Model paper to mimic how AMoC v3.0 works. We showcase the first and the last sentences with the corresponding outputs of our model (see Fig. 3). The nodes in the graph have the following colors: dark blue - text-based and active; light blue - text-based and not active; dark red - inferred and active; light red - inferred and not active. Also, note that there are maximum 10 nodes displayed, even though there might be more. This decision was made so that the concepts could be easily followed. The parameters used were the default ones, as presented in Table 1.

- First text sentence: A young knight rode through the forest
  - Top Graph Concepts: None
  - Model Generated sentences (selection of 2):
    * The girl who rode it was an apprentice knight, and she wore a light pink dress. "This one is my daughter's second bride."
    * At this moment he stopped. He looked down at the black sword which he held in his hand.
- Last text sentence: She married the knight
  - Top Graph Concepts: dragon red knight princess king
  - Model Generated sentences (selection of 2):
    * He married her by making him the best knight he could be. Because of that, his knights would be able to make the top of this league, even without the royal crown.
    * "The man in charge of the palace."

(a) First sentence

(b) Last sentence

Fig. 3. AMoC graphs after each selected sentence.

3.2 Correlations to the Landscape Model

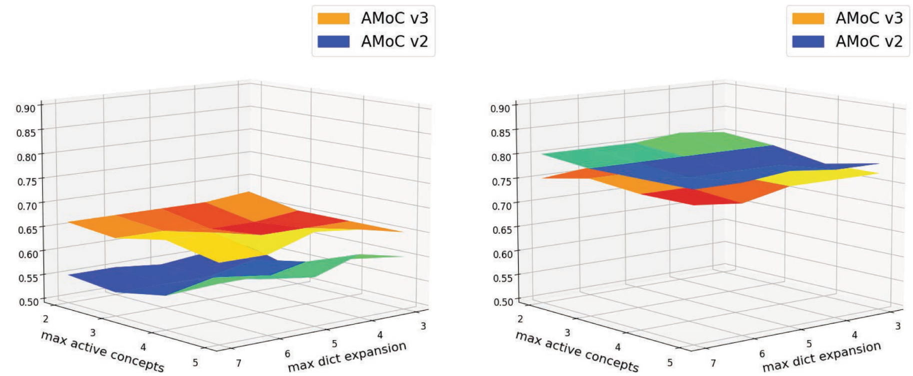
The first experiment involves analyzing the correlation between the activation scores presented in the Landscape Model paper and the activation scores obtained by AMoC v2.0 and AMoC v3.0. The scores from the Landscape Model are from 0 to 5 for each sentence, where 0 denotes that the word is not present at all in the memory, while 5 is the highest level of activation. The scores are for both text-based and inferred concepts. However, since inferring new concepts differs from method to method, we decided to consider only text-based words in this analysis.

The setup of the experiment was the following: we performed a grid-search for two parameters *maximum active concepts* and *maximum dictionary expansion*, while keeping the other parameters fixed for both models, namely: for AMoC v2.0, the *activation score* was set to 0.3; for AMoC v3.0, the *Model* was GPT-2, the *PrevSent* was 1, the *AttThresh* was 0.3, the *Imagin* was 1, the *WD* was 0.1, and the *UseAOE* was True. The *maximum active concepts* was varied between 2 and 5, and the *maximum dictionary expansion* was varied between 3 and 7. Then, for each case, the Spearman correlation between AMoC models was computed with the scores from the Landscape model. Two types of Spearman correlations were considered: correlation per word with respect to the sentences and correlation per sentence with respect to the words.

Table 3 shows the Spearman correlation scores. The scores argue that, in general, there is a correlation between the two approaches, even though they are approaching the problem from different starting points. While inspecting the word correlation, AMoC v3.0 is performing better than its previous version. AMoC v2.0 is performing slightly better in the sentence correlation category, but only with a marginal difference of .03. Figure 4 depicts surface plots with plots scores obtained from the grid search as a surface plot; the views are consistent with the mean scores.

Table 3. Mean Correlation Scores.

Spearman Correlation	AMoC v2.0	AMoC v3.0
Per Word	.566	.656
Per Sentence	.791	.764



(a) Word correlations (b) Sentence correlations

Fig. 4. Surface plots denoting correlations between both AMoC models and the Landscape Model.

An important observation is to be made about AMoC v3.0 compared to AMoC v2.0 and the Landscape model, as the last two give a high weight to the text-based words and also to their absence. For example, in the last two sentences from the text (*"The princess was very thankful to the knight. She married the knight."*), the activation score in the Landscape Model for the word "dragon" is 0, which means that the reader forgot everything about the dragon. In a similar manner, when the knight and the dragon fought, the princess also had 0 as activation. We believe that those scores should be lower, but not 0, since we are referring to the central characters of the story. As such, AMoC v3.0 offers activation of approximately 0.8 (out of 1) for "dragon" in the last sentences; nevertheless, this also has a negative impact on the overall correlations with the Landscape model results.

3.3 Diffentiating Between High-Low Cohesion Texts

This experiment was conducted to assess the extent to which the features derived from AMoC differentiate between high-low cohesion texts. Moreover, we evaluate whether cohesion differences have a lower impact on an individual with higher knowledge by changing the simulation parameters for AMoC.

The considered dataset consists of 19 texts in two forms [12], one having high cohesion and one having low cohesion. The (initial) low cohesion texts were modified by expert linguists to have a higher cohesion, but the same ideas were retained in both versions. In general, the texts are of medium length, with a mean length of approximately forty sentences. The high cohesion texts tend to be a little longer both in the number of words and of sentences, but this property did not affect the experiment.

The Linear Mixed Effects [7] statistical test was employed to analyze how well AMoC models differentiate between the same texts, one with a high and the other with low cohesion. The parameters were similar for the two models, namely: the maximum dictionary expansion was set to 9, while the maximum active concepts were set to 7. For AMoC v3.0, all the other parameters were given the default values since it supports multiple configurable parameters.

The majority of the features computed with AMoC v3.0 were statistically significant in differentiating between high-cohesion and low-cohesion texts (see Table 4). The centrality measurements are relevant to this situation because a high-cohesion text should have its concepts more tightly coupled than a low-cohesion text. In contrast, AMoC v2.0 did not perform as well - except for one feature, the others were not statistically significant. This argues for the superiority of the current version of AMoC. Also, AMoC v2.0 does not compute metrics for the active concepts, but nonetheless, the all-nodes statistics are a good indicator that the newer version is superior.

Table 4. Differences between low and high cohesion text in AMoC v2.0 and v3.0 configurations.

Feature	All/Active	AMoC v2.0		AMoC v3.0	
		F	p	F	p
Closeness centrality	active	–	–	18.37	<.001
	all	1.79	.197	6.05	.024
Betweenness centrality	active	–	–	13.06	<.001
	all	0.11	.741	0.90	.353
Degree centrality	active	–	–	10.61	<.001
	all	1.49	.237	1.95	.179
Density	active	–	–	10.61	<.001
	all	1.49	.237	1.95	.179
Modularity	active	–	–	3.71	.070
	all	4.39	.050	5.55	.029

4 Conclusions and Future Work

In this paper, we introduced the third version of the Automated Model of Comprehension. AMoC v3.0 improves on past versions by using modern Natural Language Techniques (i.e., large language models) that produce better language representation and more relevant relations between words. A detailed comparison was presented between the current version and the previous two versions, where we showed that AMoC v3.0 has improved performance and customizability compared to previous versions. These improvements enable users of AMoC to better model the reading comprehension process across texts.

Two experiments were conducted in order to evaluate AMoC and compare v2.0 to v3.0. The first experiment showed that AMoC v3.0 has higher correlations with the text-based concepts from the Landscape Model than AMoC v2.0 at the word level while maintaining a similar correlation at the sentence level. The second experiment was an analysis of the centrality graph metrics extracted from the concept graphs generated by AMoC v2.0 and AMoC v3.0 and demonstrated that AMoC v3.0 is superior. In addition, AMoC is the only existing model that simulates potential inferences drawn by the reader and provides automated predictions of readers' comprehension and text quality with no experimenter-driven decisions. The better modeling outcomes of AMoC v3.0, combined with the simulation of inferences and predicted comprehension, have implications for the design of curricula. For example, a teacher can use the generated inferences to design more personalized assessments of reading comprehension.

Future work on AMoC v3.0 will explore its capabilities by testing more complex reading outcomes, such as reading times. Additionally, we anticipate using AMoC to develop texts and reading assessments. Applying the text analysis provided by AMoC to educational settings affords the opportunity to improve the specifications of the model and assess its utility as a tool for curriculum design.

Acknowledgements. This work was supported by the Ministry of Research, Innovation, and Digitalization, project CloudPrecis, Contract Number 344/390020/06.09.2021, MySMIS code: 124812, within POC, the Institute of Education Sciences (NSF R305A130124, R305A190063), the U.S. Department of Education, and the National Science Foundation (NSF REC0241144; IIS-0735682).

References

1. Van den Broek, P., Young, M., Tzeng, Y., Linderholm, T.: The landscape model of reading: inferences and the online construction of a memory representation. The Construction of Mental Representations During Reading, pp. 71–98 (1999)
2. Brown, T., et al.: Language models are few-shot learners. In: Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901 (2020)
3. Corlatescu, D.-G., Dascalu, M., McNamara, D.S.: Automated model of comprehension V2.0. In: Roll, I., McNamara, D., Sosnovsky, S., Luckin, R., Dimitrova, V. (eds.) AIED 2021. LNCS (LNAI), vol. 12749, pp. 119–123. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-78270-2_21
4. Dascalu, M., McNamara, D.S., Crossley, S.A., Trausan-Matu, S.: Age of exposure: a model of word learning. In: 30th AAAI Conference on Artificial Intelligence, pp. 2928–2934. AAAI Press (2016)
5. Dascalu, M., Paraschiv, I.C., McNamara, D.S., Trausan-Matu, S.: Towards an automated model of comprehension (AMoC). In: Pammer-Schindler, V., Pérez-Sanagustín, M., Drachsler, H., Elferink, R., Scheffel, M. (eds.) EC-TEL 2018. LNCS, vol. 11082, pp. 427–436. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98572-5_33
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/n19-1423
7. Gałecki, A., Burzykowski, T.: Linear mixed-effects model. In: Gałecki, A., Burzykowski, T. (eds.) Linear Mixed-Effects Models Using R, pp. 245–273. Springer, New York (2013). https://doi.org/10.1007/978-1-4614-3900-4_13
8. Honnibal, M., Montani, I.: spaCy 2: natural language understanding with bloom embeddings. Convolutional Neural Netw. Incremental Parsing **7**(1) (2017)
9. Kintsch, W.: The role of knowledge in discourse comprehension: a construction-integration model. Psychol. Rev. **95**(2), 163 (1988)
10. Kintsch, W., Welsch, D.M.: The construction-integration model: a framework for studying memory for text, pp. 367–385. Lawrence Erlbaum Associates Inc., Hillsdale (1991)
11. McNamara, D.S., Kintsch, W.: Learning from texts: effects of prior knowledge and text coherence. Discourse Process. **22**(3), 247–288 (1996)
12. McNamara, D.S., Louwerse, M.M., McCarthy, P.M., Graesser, A.C.: Coh-metrix: capturing linguistic features of cohesion. Discourse Process. **47**(4), 292–330 (2010)
13. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representation in vector space. In: Workshop at ICLR (2013)
14. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)

15. Radford, A., et al.: Language models are unsupervised multitask learners. OpenAI Blog **1**(8), 9 (2019)
16. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30, pp. 5998–6008. Curran Associates Inc. (2017)
17. Vig, J.: BertViz: a tool for visualizing multihead self-attention in the BERT model. In: ICLR Workshop: Debugging Machine Learning Models (2019)