An Evaluation Model for Auto-generated Cognitive Scripts

Ahmed M. ELMougi[1], Yasser M. K. Omar[3]
College of Computing and Information Technology
Arab Academy for Science, Technology and Maritime
Transport, Cairo, Egypt

Rania Hodhod[2]
TSYS School of Computer Science
Columbus State University
GA, USA

***Abstract*—Autonomous intelligent agents have become a very important research area in Artificial Intelligence (AI). Socio-cultural situations are one challenging area in which autonomous intelligent agents can acquire new knowledge or modify existing one. Socio-cultural situations can be best represented in the form of cognitive scripts that can allow different techniques to be used to facilitate knowledge transfer between scripts. Conceptual blending has proven successful in enhancing the social dynamics of cognitive scripts, where information is transferred from similar contextual scripts to a target script resulting in a new blended script. To the extent of our knowledge, there is no computational model available to evaluate these newly generated cognitive scripts. This work aims to develop a computational model to evaluate cognitive scripts resulting from blending two or more linear cognitive scripts. The evaluation process involves: 1) using the GloVe similarity to check if the transferred events conceptually fit the target script; 2) using the semantic view of text coherence to decide on the optimal position(s) to place the transferred event(s) in the target script. Results show that the GloVe similarity can be applied successfully to preserve the contextual meaning of cognitive scripts. Additional results show that GloVe embedding gives higher accuracy over Universal Sentence Encoder (USE) and Smooth Inverse Frequency (SIF) embedding but this comes with a high computational cost. Future work will look into reducing the computational cost and enhancing the accuracy.**

Keywords*—Autonomous intelligent agents; socio-cultural situations; cognitive scripts; conceptual blending; contextual structural retrieval algorithms; text coherence; sentence embedding

I. Introduction

Autonomous intelligent agents are a very important research area in Artificial Intelligence (AI). Intelligent agents possessing mental abilities, such as knowledge, belief, intention, and obligation can have human-like capabilities, such as artificial intuition and imagination, analogy and conceptual blending, design, writing poetry, argumentation, dialogue generation, negotiation abilities and shared mental models. It is important for autonomous intelligent agents to be able to acquire new knowledge or modify existing one. This seems to be quite difficult in some domains, such as socio-cultural situations because of the temporal and causal relations twined in these situations. Generally, people think of a situation as a sequence of routine actions/events that can be represented in the form of cognitive scripts. These events are connected temporally or causally with preceding and succeeding events [1].

It is a challenge to develop an intelligent agent that has the mechanism to change the knowledge it has and learn from external knowledge. Humans use analogical reasoning [2] to learn by simply transferring knowledge from a more familiar situation to a less familiar one making use of the structural similarity of the two situations. Conceptual blending is a theory of cognition, developed by Gilles Fauconnier and Mark Turner [3], that uses analogical reasoning to enhance the social dynamics of one script (target) by transferring events from a contextually similar script (base) resulting in a new blended script [1], [4], [5], [6].

One shortcoming that can be seen in these works is the fact that the evaluation of the resulting scripts needs human intervention; there should be a computational model to evaluate the newly generated blended scripts particularly in real time applications such as interactive narrative applications [4], [5]. Some events may be transferred to the target script while they don’t conceptually fit it. For example, the event “audience listens to movie” may be transferred to the cinema script when blending it with the lecture script. An event may be inserted into an unlogic position in the target script. For example, the event “light on” may be inserted before the event “movie starts” in the cinema script, when blending it with the lecture script, while it must be inserted after the event “movie ends”.

This work aims to develop a computational evaluation model for blended linear cognitive scripts. The approach used in [1] and [6] is adopted in this work to select events to be transferred from the base script to the target script. The evaluation process is two phases: Firstly, checking if the selected events can be added to the target script; The GloVe similarity ratio between every selected event and the target script is computed. If this ratio exceeds or equals a specified threshold, the event can be added to the target script. Secondly, using text coherence evaluation techniques is to specify the optimal position(s) to insert the selected event(s). The target script is converted into a text with every event converted into a sentence. The transferred events are converted into sentences. The optimal positions of the transferred events are the ones with the highest semantic text coherence [7]; in this technique, every sentence is converted into a vector and the semantic similarities between every two subsequent sentences are computed and then averaged to compute the coherence of the text. The semantic similarity between two sentences is the cosine similarity between their corresponding embedding vectors.

This paper is organized as follows: Section II gives background about cognitive scripts and analogical reasoning. Section III presents related work in the field of enhancing the dynamics of socio-cultural situations highlighting approaches in the fields of text coherence evaluation techniques and sentence embedding. Section IV introduces the proposed model. Section V shows results discussion. Section VI provides conclusion. Section VII provides suggestions for future work.

II. Background

A. Cognitive Scripts

We live in a world consisting of objects and events that relate these objects to each other. People store their knowledge about socio-cultural situations as a sequence of events, such as “Entering a restaurant” or “Attending a lecture” situations. Such socio-cultural situations are best represented in the form of cognitive scripts with events connected by directional edges. The events are either temporally or causally connected in a way that defines the context of a cognitive script. A cognitive script may be linear or multi-branched as shown in Fig. 1 in which each path in the multi-branched script can be seen as an independent linear script [1]. In this figure, the cinema cognitive script consits of different events such as “Audience buys ticket”, “Lights off”, and “Audience watches movie”. These events are conneted to their preceding and succeeding events by directional edges. The script consists of four different paths, each path represent a linear script. These paths have three interscting events; “Audience enters auditorium”, “audience watches movie” and “Movie ends”.

B. Analogical Reasoning

Analogical reasoning is a core process in human cognition defined as the ability to perceive and use relational similarity between two situations. In analogical reasoning, the relational similarity between two situations can be used to make inferences from one situation to the other. The first situation is called the base situation and is more familiar than the second situation which is called the target situation [2].

Analogical reasoning can be applied to production rules, cases, semantic networks, and cognitive scripts [8] and is usually comprised of three stages; retrieval, mapping, and evaluation. Retrieval is the process of retrieving a situation from long-term memory that is analogous to a situation in working memory. Mapping is the core process in analogical reasoning and is defined as the process of finding structural similarity between two situations and making inferences from a base situation to a target situation. Two situations can be structurally similar if there is an alignment between the two situations according to their structural similarity. Only then projected inferences from a base situation to a target situation can occur noting that every object in the base situation must be aligned to only one object in the target situation. This is known as one-to-one-correspondence. Sterman and his colleagues at MIT made an interesting analogy between the inflow and outflow of water in a bathtub with CO_2 emissions and removal in the atmosphere. In this analogy, the bathtub corresponds to the atmosphere. Water inflow and water outflow correspond to CO_2 emissions into the atmosphere and CO_2 removal respectively. Another requirement for structural consistency of two situations is that when two relations are matched, their arguments must be matched. Finally, inference from the base situation to the target situation is selective. People prefer to infer relations that are consistent with the matching structure of the two situations, in addition to using the systematicity principle. Lastly, evaluation takes place where analogy and inferences are accepted or rejected. Three factors affect the evaluation: The first factor is factual correctness that clarifies whether inferences are true or not. This may be incorrect in the case of future predictions. Another aspect related to factual correctness is adaptability which means that inferences can be accepted if they can be adapted easily in the target situation. The second factor is goal relevance and is important in problem-solving situations. The third factor is related to whether new knowledge can be added to the target situation or not. This may be risky, but it is important in brainstorming or unfamiliar situations [2].

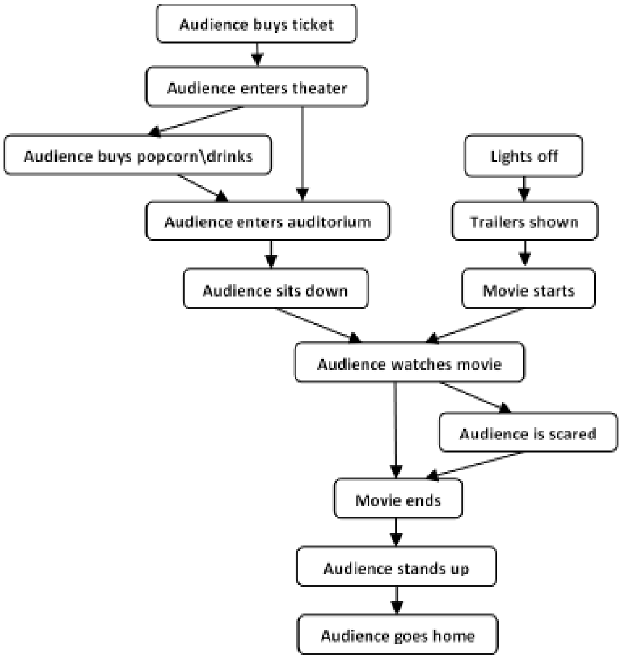


Fig. 1. Multi-Branched Cinema Cognitive Script.

III. Literature Review

A. Related Work

Many works have been done to create new knowledge from existing one in socio-cultural situations.

Hodhod and Magerko (2014) tried to give AI improvisational agents the ability to improvise new non-traditional scenes making use of existing social cognitive scripts [4]. Improvisational acting is a creative process implemented by actors on stage in real time. In this process, actors use their perceptions of the environment to create stories with each other. The authors developed the Pharaoh algorithm that retrieves a contextually similar cognitive script (base) to be blended with a target script based on the events’ appearances in the scripts and their least common parents [9]. iPharaoh, a modification of the Pharaoh algorithm, was after then developed to enhance the performance of the Pharaoh algorithm in terms of precision and recall, in addition to reducing the retrieval time [6]. In this work, a target script is chosen from the script-base. The Pharaoh algorithm is then used to find the highest contextually similar script among the script-base, referred to as the base script. The same conceptual blending rules used in the cognitive system, Sapper [10], are applied to the target and base scripts. These rules keep the structure of the target script and allow the addition of new events from base script or semantic networks.

The main drawbacks in Pharaoh and iPharaoh are that both algorithms rely on exact matching, in addition to the absence of a computational evaluation model; the blended scripts are evaluated by humans.

Permar and Magerko (2013) used another approach in [5]. The authors were interested in scripts in the domain of pretend play. The scripts are represented using a Directed Acyclic Graph (DAG). The authors used a blending algorithm that consists of three phases; counterpart mapping, mapping selection, and mapping application. Node mapping in the counterpart phase is a key process, the priority for node mapping followed the following rules:

- If the path of the target script has an iconic node, this node must be replaced.
- If the path of the base script has an iconic node, it has a priority to replace the mapped node in the target script.
- All pairs of paths are ordered according to the number of mapped nodes from the highest to the lowest.
- If a node is selected in a path, it will not be selected in the succeeding paths.

Some of the drawbacks in this approach are the use of exact matching, blending can take place between two scripts only, causality is not considered, and blended scripts are evaluated by humans.

Gawish et al. (2013) modified Pharaoh to allow the use of WordNet for lexical similarity [1]. WordNet is a lexical database with nouns and verbs organized into hierarchies of an *is–a* relation [11]. This representation makes WordNet particularly suited for similarity measures between two distinct but similar words. The model used consists of four blocks; evolved script-base, retrieval module, commonsense knowledge base, and learning module. The retrieval module retrieves the base script, which is the one that has the highest contextual similarity with the target script. The learning module uses two evolutionary processes to create a blended script; crossover and mutation. Crossover is used to insert new events or connections from the base script into the target script. Mutation is used to insert new events or connections from ConceptNet. ConceptNet is a large-scale commonsense semantic network of assertions of commonsense knowledge that represents the spatial, physical, social, temporal, and psychological aspects of everyday life [12]. Although this work addressed some of the drawbacks in the previous works, such as the use of lexical similarity and the ability to learn from other scripts as well as commonsense knowledge, it still did not address the missing ability of automatic evaluation of the resulting blended scripts.

B. Text Coherence Evaluation Techniques

Thinking of a cognitive script as a sequence of text can be the starting point to allow the emergence of an automated evaluation model. Testing coherence is to specify if the text is well written or not can be used to evaluate cognitive scripts. Text coherence evaluation has been used in many applications, such as machine translation, text generation, and summarization. Two important approaches in text coherence evaluation are introduced in [7]. The first approach is based on the syntactic view of text coherence which considers the change of the syntactic role of the text entities through adjacent sentences based on the Centering Theory. This theory asserts that texts in which successive statements mention the same entities are more coherent than texts in which multiple entities are mentioned.

The other approach relies on the semantic view of text coherence which implies that coherent text has high lexical cohesion between its sentences. This means that subsequent sentences have high semantic similarity. The semantic similarities between every two subsequent sentences are measured and then averaged to get the text coherence. This is illustrated in (1).

$$coherence(T) = \frac{\sum_{i=1}^{n-1} sim(S_i, S_{i+1})}{n-1} \tag{1}$$

Where $sim(S_i,S_{i+1})$ is the measure of semantic similarity between sentences Si and Si_{+1}.

The authors experimented with three different approaches to measure the semantic similarity between two sentences. The first approach measures the semantic similarity of two sentences in terms of word overlap. This is illustrated in (2).

$$sim(S_1, S_2) = \frac{2|words(S_1) \cap words(S_2)|}{(|words(S_1)| + |words(S_2)|)} \tag{2}$$

Where $words(S_i)$ is the set of words in sentence i. The main drawback of this approach is that two sentences may have no common words, but they are semantically related. For example, the sentences “game ends” and “audience stands up” have no common words although they are semantically related.

The second approach is to use WordNet similarities between words in the two sentences. Since the WordNet similarity between words is dependent on the meaning of the words, the higher the similarity value between two words is, the more similar these words are. This is illustrated in (3):

$$sim(S_1, S_2) = \frac{\sum_{\substack{w_1 \in S_1 \\ w_2 \in S_2}} \underset{\substack{c_1 \in senses(w_1) \\ c_2 \in senses(w_2)}}{argmax} sim(c_1, c_2)}{|S_1||S_2|} \tag{3}$$

Where $|S_i|$ is the number of words in sentence i. Since the appropriate senses of words w_1 and w_2 are not known, the similarity measure will select the senses which will maximize $sim(c_1,c_2)$.

One concern in this approach is the possibility of the words with high WordNet similarity to be irrelevant to the context/meaning of the text in the cognitive scripts.

The third approach follows the method in [13] and converts every sentence into an embedding vector then it measures the cosine similarity between the two embedding vectors. This approach is the approach used in the evaluation model used in this work.

C. *Sentence Embedding*

Different word and sentence embeddings have been developed to encode words and sentences as numerical vectors to be used in different natural language processing applications. Examples of these embeddings which will be used in this work are GloVe, Universal Sentence Encoder (USE), and Smooth Inverse Frequency (SIF).

GloVe embedding works on the word level. It uses a specific weighted least square model that uses a global word-word co-occurrence matrix for training to make efficient use of statistics. This co-occurrence matrix is constructed by training different corpora [14]. The GloVe vector of a sentence is the average of the GloVe vectors of its words.

USE works on the word, sentence, and paragraph levels where any word, sentence, or paragraph is converted into a vector of 512 dimensions. This encoder uses two different models. One makes use of the transformer architecture. This model achieves higher accuracy with greater model complexity and resource consumption. The other model is implemented as a Deep Average Network (DAN) which achieves efficient inference with slightly reduced accuracy [15].

SIF provides a new simple sentence embedding technique. This technique computes the weighted average of the word vectors in the sentence and then removes the projections of the average vectors on their first singular vector. The weight of a word w is *a / (a + p(w))* with *a* being a parameter and *p(w)* the estimated word frequency [16].

IV. Proposed Model

Our proposed model focuses on linear cognitive scripts in which events are provided in the form of “subject/verb/object”. For example, “audience enters stadium”, “audience thanks lecturer”, or “waiter delivers menu”. Events can also be provided in the form of “subject/verb”. For example, “movie starts”, “game ends”, or “lecture ends”. Finally, events can be in the form of “subject/adjective”. For example, “light on”, “light off” or “audience excited”. The subject and object of an event represent the event parameters and the verb or adjective represents the event action.

Before applying the evaluation model, some data processing must be executed. All scripts are converted into texts with every event converted into a sentence. For every script text, the average GloVe similarity is computed and stored. This is attained by computing the GloVe similarity of every sentence of the script with the remaining sentences of the script and then averaging these computed similarities. The pseudo code to compute he GloVe similarity between a sentence s and a sequence of sentences ss is shown in Fig. 2.

An important note to be considered is the assumptions that opposite events don’t exist in direct sequence in most cases. Opposite events are these events that have the same parameters but have opposite actions. For example, “light on” and “light off”, “audience enters stadium” and “audience leaves stadium”, and “movie starts” and “movie ends” don’t exist in direct sequence in most cases. All sequences of the blended script having subsequent opposite events are rejected when specifying the optimal positions of the events added to the target script. All opposite actions are stored before applying the evaluation model.

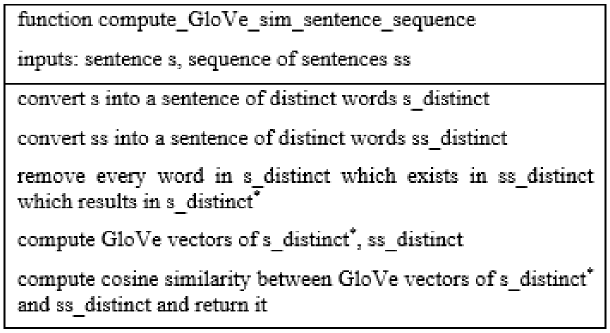
A. *First Evaluation Phase*

The first evaluation phase evaluates if the events selected from the base script can be added to the target script without changing its context. Since the GloVe embedding uses a global word-word co-occurrence matrix, it can be used to evaluate the probability of the existence of a sentence in the context of other sentences. When an event is selected from the base script, it is converted into a sentence and the GloVe similarity between this sentence and the target script text is computed and then divided by the average GloVe similarity of the target script text. The resulting value is defined as the GloVe similarity ratio between the event and the target script. If this ratio exceeds or equals a specified threshold, the selected event can be added to the target script.

B. *Second Evaluation Phase*

The second evaluation phase is used to specify the optimal positions to insert the events transferred from the base script into the target script. This can be done using text coherence evaluation techniques. The target script is converted into a text and the transferred events are converted into sentences. The optimal positions of the transferred events achieve the highest text coherence.

Text coherence evaluation techniques relying on the syntactic view of text coherence have a serious drawback when applied to cognitive scripts used in this work. This drawback is the reliance on the entities of the text while ignoring verbs and adjectives. Verbs and adjectives are very important in cognitive scripts because they represent the events’ actions while entities represent the events’ parameters. For example, consider the two events in the cinema script; “light on” and “light off”. These sentences have the same entity “light”. The entity’s role in the two sentences is a subject. If these events are exchanged, the coherence of the script text will not change although the logic of the text is completely different. The same problem occurs with events such as, “audience enters theater” and “audience leaves theater”.



```
function compute_GloVe_sim_sentence_sequence
inputs: sentence s, sequence of sentences ss

convert s into a sentence of distinct words s_distinct
convert ss into a sentence of distinct words ss_distinct
remove every word in s_distinct which exists in ss_distinct
which results in s_distinct*
compute GloVe vectors of s_distinct*, ss_distinct
compute cosine similarity between GloVe vectors of s_distinct*
and ss_distinct and return it
```

Fig. 2. Pseudo Code to Compute the GloVe Similarity between a Sentence and a Sequence of Sentences.

The technique used in this work relies on the semantic view of text coherence and uses (1) where the similarity between any two subsequent sentences is the cosine similarity between their corresponding embedding vectors.

V. Results and Discussion

A. Dataset

To the extent of our knowledge, there is no benchmark dataset for cognitive scripts. Four linear cognitive scripts adopted from [6] are used in this experiment; stadium, lecture, restaurant, and cinema. They were chosen because they provide a good variation from less detailed scripts such as stadium and lecture scripts to more detailed scripts such as restaurant and cinema scripts. The four scripts as converted into texts are shown in Fig. 3, Fig. 4, Fig. 5, and Fig. 6. In these figures, every sentence of the text represents an event of the corresponding cognitive script.

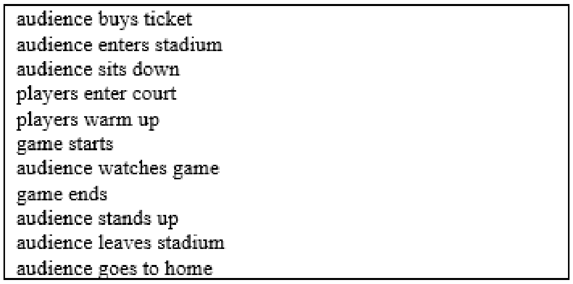
These four cognitive scripts are used to create a dataset to compute the GloVe similarity ratio threshold, evaluate the performance of the first evaluation phase, and evaluate the performance of the second evaluation phase. The procedure for creating the dataset is explained as follows:

- Every script of length n is split into all possible partitions of lengths n-1 and 1. The “n-1” events partition will represent a target script and the other partition will represent an event selected to be transferred to this target script.
- Again, every script of length n is split into all possible partitions of lengths n-2 and 2. The “n-2” events partition will represent a target script and the other partition will represent two events selected to be transferred to the target script.

Since this work focuses on enhancing the social dynamics of one cognitive script by transferring events from a contextually similar cognitive script, it is assumed that the core of the target script exists and at most two events are transferred from the base script to this target script.

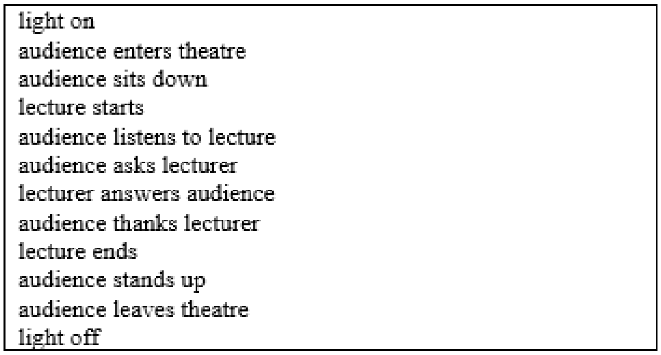
For a script of length n, the number of all possible partitions of lengths n-1 and 1 is n. Similarly, the number of all possible partitions of lengths n-2 and 2 is C^{n}_{2}. The four scrips will create a dataset of 340 script instances; 50 “n-1/1” script instances and 290 “n-2/2” script instances.

It is worth noting that “n-1” and “n-2” scripts are converted into texts and their average GloVe similarities are computed and stored.



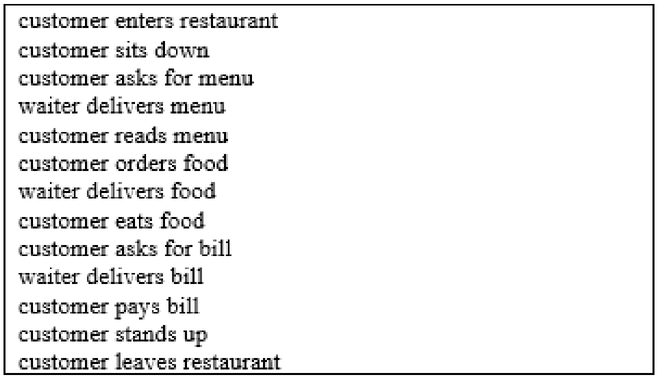
audience buys ticket
audience enters stadium
audience sits down
players enter court
players warm up
game starts
audience watches game
game ends
audience stands up
audience leaves stadium
audience goes to home

Fig. 3. The Stadium Linear Cognitive Script as Converted into Text.



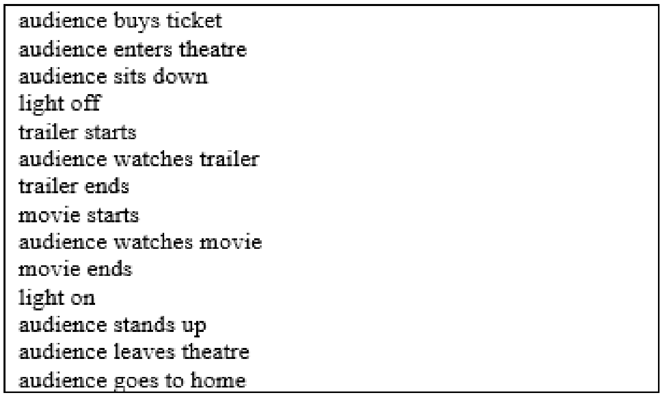
light on
audience enters theatre
audience sits down
lecture starts
audience listens to lecture
audience asks lecturer
lecturer answers audience
audience thanks lecturer
lecture ends
audience stands up
audience leaves theatre
light off

Fig. 4. The Lecture Linear Cognitive Script Converted into Text.



customer enters restaurant
customer sits down
customer asks for menu
waiter delivers menu
customer reads menu
customer orders food
waiter delivers food
customer eats food
customer asks for bill
waiter delivers bill
customer pays bill
customer stands up
customer leaves restaurant

Fig. 5. The Restaurant Linear Cognitive Script Converted into Text.



audience buys ticket
audience enters theatre
audience sits down
light off
trailer starts
audience watches trailer
trailer ends
movie starts
audience watches movie
movie ends
light on
audience stands up
audience leaves theatre
audience goes to home

Fig. 6. The Cinema Linear Cognitive Script Converted into Text.

B. Computing the GloVe Similarity Ratio Threshold

The proposed model uses GloVe vectors of 300 dimensions that are created by training Common Crawl (840B tokens, 2.2 M vocab, cased, 2.03 GB download). These vectors can be downloaded from https://github.com/stanfordnlp/GloVe.

For all “n-1/1” instances, the GloVe similarity ratio between the event and the “n-1” script is computed. For all “n-2/2” instances, the GloVe similarity ratio between every event of the two events and the “n-2” script is computed. The GloVe similarity ratio of some events can’t be computed since the parameters and actions of these events already exist in the target scripts. Examples of such events are “trailer starts”, “audience watches movie”, and “movie ends” when added to the cinema script.

Six different actions are chosen to be added to the "n-1" and "n-2" target scripts. Since the approach used for selecting events to be transferred to the target script implies parameter mapping [6], only the GloVe similarity ratios between these actions and the target scrips are computed. The selected actions are "sleeps", "eats", "listens', "teaches", "sings", and "dances". If the action exists in one of the target scripts, its GloVe similarity ratio will not be computed.

Human intervention is essential to decide for every action if it can be added or not to the target script. This human intervention focuses on cases where there is a certainty about accepting or rejecting the action. Cases, where there is an uncertainty about acceptance or rejection, are excluded. This is shown in Table I.

The script name which is underlined indicates that the action already exists in the corresponding script and its GloVe similarity ratio will not computed. From Table I, there is an uncertainty about some actions whether they are accepted or rejected in some target scripts. For example, there is an uncertainty about adding the action "listens" to the restaurant script since people may listen to music while eating. Similarly, people may eat while watching a game in the stadium or watching a movie in the cinema. The GloVe similarity ratios between the six actions and all "n-1" and "n-2" scrips are computed. These results are combined with the results of computing the GloVe similarity ratios using the "n-1/1" and "n-2/2" scripts and are used to specify the GloVe similarity ratio threshold which is empirically set to 0.8.

C. *Evaluation of the First Evaluation Phase*

The GloVe similarity ratio threshold specified empirically above is used to deduce the confusion matrix for this phase, which is shown in Table II, where True Positive (TP) = 507, True Negative (TN) = 1392, False Positive (FP) = 139, and False Negative (FN) = 60. Performance of this phase will be measured in terms of sensitivity, specificity, and accuracy. Sensitivity is calculated as *TP * 100 / (TP + FN)*, specificity is calculated as *TN * 100 / (TN + FP)*, and accuracy is calculated as *(TP + TN) * 100 / (TP + TN + FP + FN)*. Sensitivity = 89.42%, specificity = 90.92%, and accuracy = 90.51%.

TABLE. I. ACCEPTANCE/REJECTION OF THE SIX ACTIONS WITH RESPECT TO THE FOUR SCRIPTS

Action	must be accepted in	must be rejected in
sleeps		Stadium, lecture, restaurant, and cinema
eats	Restaurant	Lecture
listens	Lecture	Stadium and cinema
teaches	Lecture	Stadium, restaurant, and cinema
sings		Stadium, lecture, restaurant, and cinema
dances		Stadium, lecture, restaurant, and cinema

TABLE. II. CONFUSION MATRIX OF THE FIRST EVALUATION PHASE

	GloVe similarity ratio >= 0.8	GloVe similarity ratio < 0.8
The event must be added to the target script	507	60
The event must not be added to the target script	139	1392

D. *Evaluation of the Second Evaluation Phase*

The transferred events are inserted into all possible positions in the target script. The coherences of all these texts are computed. The optimal positions of the transferred events are these positions that result in the highest coherence of the blended script text. The accuracy is defined as the percentage of the blended script texts rather than the optimal blended text that have coherence lower than to that of the optimal blended text.

Three sentence embeddings will be used and compared; GloVe, USE, and SIF. USE will be implemented by a light weighted version of the transformer model which can be used with limited computation resources but still gives good performance. This can be viewed at https://tfhub.dev /google/universal-sentence-encoder-lite/2. The code for computing the cosine similarity between two SIF sentence embeddings can be viewed at https://www.kaggle.com/ procode/sif-embeddings-got-69-accuracy.

To evaluate this phase, two cases will be tested:

- For all "n-1/1" instances, the event is added to the "n-1" target script.
- For all "n-2/2" instances, the two events are added to the "n-2" target script simultaneously.

Adding one event in all possible positions in an "n-1" target script will result in n blended texts. One of them is the optimal text. Accuracies are computed for all "n-1/1" instances of every script and then averaged for every one of the three sentence embeddings used. The accuracies of this case are shown in Table III and Fig. 7. The accuracy of every sentence embedding technique is the average of its accuracies for the four scripts. The accuracy for adding one event to a target script of length n-1 using GloVe is 86.52%, USE is 75.51%, and SIF is 72.23%.

TABLE. III. ACCURACIES OF ADDING ONE EVENT TO "N-1" TARGET SCRIPTS

Script	Accuracy of GloVe%	Accuracy of USE%	Accuracy of SIF%
Stadium	80.99	85.95	71.07
Lecture	88.19	70.14	59.72
Restaurant	84.02	63.31	81.07
Cinema	92.86	82.65	77.04

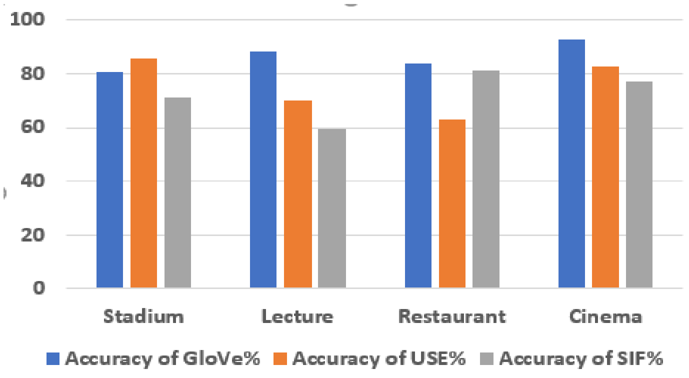


Fig. 7. Accuracies of Adding One Event to "n-1" Target Scripts.

Adding two events simultaneously in all possible positions in an "n-2" target script will result in P^n_2 blended texts. One of them is the optimal text. Accuracies are computed for all "n-2/2" instances of every script and then averaged for every one of the three sentence embeddings used. The accuracies of this case are shown in Table IV and Fig. 8. The accuracy of every sentence embedding technique is the average of its accuracies for the four scripts. The accuracy for adding two events to a target script of length n-2 using GloVe is 92.54%, USE is 79.02%, and SIF is 74.75%.

The GloVe embedding achieves the highest accuracy for the two cases. The accuracy of the second evaluation phase is the average of the accuracies of the two cases using GloVe embedding. The accuracy of the second evaluation phase is 89.53%. The overall accuracy of the evaluation model is the product of the accuracies of the first evaluation phase and the second evaluation phase. The overall accuracy of the evaluation model is 81.03%.

Another interesting metric to evaluate the proposed model is to study the effect of exchanging two events that share the same parameters but have different actions. In entity-based text coherence evaluation techniques, exchanging such events does not change the coherence of the script text although the logic of the script is completely different. The four scripts used in this work contains 14 event pairs of such type. They were exchanged such that only two events are exchanged per one time resulting in 14 different scripts. Coherences of these 14 scripts' texts were computed and then compared to the four scripts' optimal script texts' coherences using the three sentence embeddings used in this work. For Glove embedding, 9 of these scripts had coherence lower than that of the optimal scripts. For USE, 5 of these scripts had coherence lower than that of the optimal scripts. For SIF embedding, 3 of these scripts have coherence lower than that of the optimal scripts. GloVe embedding achieved higher accuracy over USE and SIF in the case of exchanging two events that share the same parameters but have different actions.

E. Discussion

The evaluation model achieves a promising accuracy but with a high computational cost.

Transferring two events to a target script of length n-2 simultaneously requires P^n_2 i.e. n*(n-1) computations of text coherence to decide on the optimal blended script while, transferring them sequentially requires n-1 computations for the first event and n computations for the second event with a total number of 2n-1 computations. Future work should focus on reducing the computational cost and enhancing the accuracy.

TABLE. IV. ACCURACIES OF ADDING TWO EVENTS TO" N-2" TARGET SCRIPTS SIMULTANOUSLY

Script	Accuracy of GloVe%	Accuracy of USE%	Accuracy of SIF%
Stadium	88.76	91.54	69.06
Lecture	93.5	75.83	57.7
Restaurant	90.87	62.44	92.51
Cinema	97.04	86.25	79.72

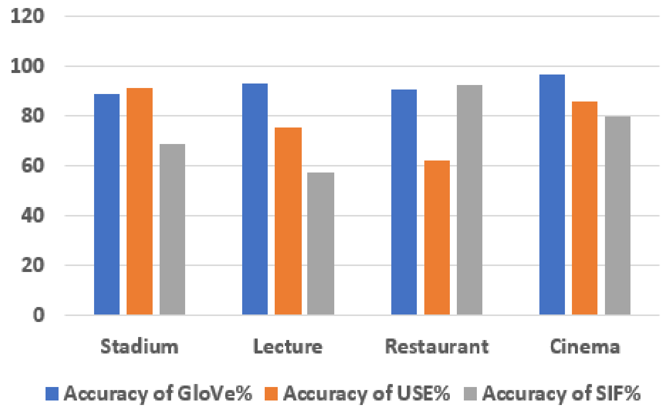


Fig. 8. Accuracies of Adding Two Events to "n-2" Target Scripts Simultanously.

The usage of a global word-word co-occurrence matrix can help explaining the reasons behind the GloVe embedding being successful in the first and second evaluation phases.

The accuracies of the four scripts using the Glove embedding ordered from the highest to lowest are that of cinema, lecture, restaurant, and stadium scripts. An explanation for the cinema script to be on top of the list is the fact that it has a lot of details and 5 pairs of start/end events. These events are "audience enters theatre" and "audience leaves theatre", "audience sits down" and "audience stands up", "light on" and "light off", "trailer starts" and "trailer ends", and "movie starts" and "movie ends". Although the restaurant script has more details than the lecture script, the lecture script has more start/end pair events than the restaurant script has. This can explain why the accuracy of the lecture script is higher than that of the restaurant script. In general, two factors, that seem to affect the accuracy of a script using GloVe embedding, are details included in the script and the number of start/end pair events the script contains.

VI. CONCLUSION

This paper introduces a computational model for evaluating blended cognitive scripts resulting from transferring new events to a target script from another cognitive script known as the base script. The proposed model focuses on linear cognitive scripts consisting of events in the form of "subject/verb/object", "subject/verb", or "subject/adjective". Subjects and objects are called the event parameters, while verbs or adjectives are called the event actions. Before an event is transferred from a base script to a target script, its parameters are mapped.

Four scripts are adopted and used to create a dataset of 340 script instances.

The evaluation process consists of two phases. The first evaluation phase evaluates if the selected events can be added to the context of the target script. The target script is converted into a text with every event converted into a sentence and the selected events are converted into sentences. The GloVe similarity ratio between every selected event and the target script is computed. If this ratio exceeds or equals a specified threshold, the event is transferred to the target script. The

GloVe similarity ratio threshold was empirically computed to be 0.8. The first evaluation phase achieved an accuracy of 90.51%.

The second evaluation phase is used to specify the optimal positions to insert the transferred events into the target script. The idea of the second evaluation phase is that the optimal positions of the transferred events are the positions that achieve the highest coherence of the blended script text. The text coherence evaluation technique used relies on the semantic view of text coherence where every sentence of the text is converted into an embedding vector, the similarities between every two subsequent sentences are computed as the cosine similarity of their embedding vectors, and then these similarities are averaged to compute the text coherence. Three sentence embeddings are used and compared. The GloVe embedding achieved higher accuracy over USE and SIF embeddings. The accuracy of the second evaluation phase is 89.53% using GloVe embedding.

The proposed model achieved an overall promising accuracy of 81.03% but with a high computational cost.

VII. Future Work

Future work will focus on reducing the computational cost and enhancing the accuracy. Different approaches may be suggested. One approach is to test the addition of two events sequentially firstly, without any precedence of one event over the other and secondly, with applying a precedence rule in transferring the two events such as transferring the event with the higher GloVe similarity ratio first.

Another approach is to convert the second evaluation phase into an optimization problem. The second evaluation phase searches for the optimal positions to insert the transferred events. Optimization techniques such as Genetic Algorithms (GA) or Particle Swarm Optimization (PSO) may be used.

With the two previously mentioned approaches, the text coherence of the blended script text may be measured using deep learning techniques such as using a Convolutional Neural Network (CNN) [17].

Converting the second evaluation phase into a problem of sentence ordering may be a solution. Recurrent Neural Networks (RNNs) may be used in this approach [18].

References

[1] M. Gawish, S Abbas, M. G. Mostafa, and A. B. M. Salem, "Learning cross-domain social knowledge from cognitive scripts," Proceedings of the 8th International Conference on Computer Engineering & Systems (ICCES), IEEE, 2013.

[2] D. Gartner and L Smith, Analogical reasoning, Encyclopedia of Human Behavior, 2nd ed., Elsevier, pp. 130–136, 2012.

[3] G. Fauconneit and M. Turner, "Conceptual integration networks," Cognitive Sciences, vol. 22, no. 2, pp. 133–268, March 1998.

[4] R. Hodhod and B. Magerko, "Pharaoh: conceptual blending of cognitive scripts for computationally creative agents," Proceedings of the Twenty-Seventh International FLAIRS Conference, May 2014.

[5] J. Permar and B. Magerk, "A conceptual blending approach to the generation of cognitive scripts for interactive narrative," Proceedings of AIIDE, 2014.

[6] M. Gawish, Developing an intelligent computation model for human episodic learning, Master thesis, Ain Shams University, May 2017.

[7] M. Lapata and R. Barzilay, "Automatic evaluation of text coherence: models and representations," Proceedings of the 19th International Joint Conference on Artificial intelligence, pp. 1085–1090, 2005.

[8] A. B. M. Salem and M. Gawish, "Study on analogical reasoning methodologies for developing analogical learning systems," International Journal of Circuits and Engineering, vol. 1, pp. 54–61, 2016.

[9] R. Hodhod, B. Magerko, and M. Gawish, "Pharaoh: context-based structural retrieval of cognitive scripts," International Journal of Information Retrieval Research (IJIRR), vol. 2, no. 3, pp. 58–71, 2012.

[10] T. Veale and D. O'Donoghue, "Computation and blending," Cognitive Linguistics, vol. 11, no. 3-4, pp. 253–281, 2000.

[11] T. Pederson, S. Patwardhan, and J. Michelizzi, "WordNet:: similarity: measuring the relatedness of concepts," Demonstration Papers at HLT-NAACL, pp. 38–41, 2004.

[12] H. Liu and P. Singh, "ConceptNet—a practical commonsense reasoning tool-kit," BT technology Journal, vol. 22, no.4, pp. 211–226, 2004.

[13] P. Foltz, W. Kintsch, and T.K. Landauer, "The measurement of textual coherence using latent semantic analysis," Discourse Processes, vol. 25, no. 2-3, pp. 285–307, 1998.

[14] Pennington, R. Socher, and C. D. Manning, "GloVe: global vectors for word representation," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543, 2014.

[15] D. Cer et al., "Universal sentence encoder," ArXiv, 1803.11175, 2018.

[16] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," Proceedings of ICLR, 2017.

[17] B. Cui, Y. Li, Y. Zhang, and Z. Zhang, "Text coherence analysis based on deep neural network", Proceedings of ACM on Conf. Info. Know. Manag. (CIKM), pp. 2027 – 2030, Nov. 2017.

[18] L. Logeswaran, H. Lee, and D. Radev, "Sentence ordering and coherence modeling using recurrent neural networks", Proceedings of AAAI, 2018.