# Cloze Story Task

**Francesco Saverio Varini** **Roberta Huang** **Mélanie Gaillochet** **Costanza Calzolari**

## Abstract

Learning of commonsense knowledge has become an important focus of research in natural language understanding in the past few years. In particular, the Story Cloze Test is a framework that was developed for evaluating story understanding and script learning. Given the first 4 sentences of a story, the system must choose the correct ending between two possible last sentences.
This paper describes our group's submission for the Story Cloze Task. We decided to work with different kinds of neural networks: convolutional neural networks (CNN), recurrent neural networks in particular long-short term memory (LSTM) networks and feed-forward neural network (FFNN). Each model was trained on an "augmented" dataset - training dataset to which we added wrong endings, since we were only provided with the correct ending. Analyzing the models' performance, we obtained that the feed-forward neural network (FFNN) provided the best result, with an accuracy of 73.4% on the validation set and of 68.8% on the test set, while training on the validation set.

## 1 Introduction

Research in natural language understanding has recently focused on learning narrative representations. At the center of several lines of research has been the Story Cloze Test where, given the first 4 sentences of a 5-sentence long story and two possible endings, the task is to choose the right (most coherent) one.

In the Story Cloze Test, the challenge is especially notable in understanding correlational relationships between events. While human performance achieves 100% accuracy, the majority baseline performances on a given test set yield an accuracy of 51.3% [1]. However, the best result was given by a model which achieved an accuracy of 75.2% on the task.

There are several complications when dealing with narrative understanding in the Story Cloze Task. The first one involves the lack of wrong endings in the training data. Indeed, to train the system in the Story Cloze task context, one needs to either use solely the validation set or create wrong endings for the training set. Several groups of researchers have opted for the former, although many methods have also been proposed to generate wrong ("negative") endings [3]. Since the validation set was relatively small, we decided, in our case, to create negative endings in order to avoid overfitting when training the model. Moreover, it has been shown in previous papers that being able to include sentiment analysis in the model really improves its performance [12][13][14]. One of the models that we developed (the feed forward neural network) thus included results for sentiment analysis as an input.

This paper proposes several models for tackling the Story Cloze Test. We will then analyze their characteristics and performance and discuss the obtained results.

## 2 Preprocessing

We decided to address the task by starting with preprocessing of the data in order to make it more understandable for the models. For CNN and SiameseLSTM, punctuations were removed and words

were lemmatized using nltk package. Then we padded the stories in order for them to have the same length. Depending on the model used, the padding was done for either the overall story, or for the context and ending separately. We also decided to PoS tag all the words and use as input for the models the tuples comprised of the word itself and its tag. While for the feed-forward neural network, we input raw sentences which are then embedded with skip-thought embeddings.

Data augmentation was then performed using different strategies. We first started out changing the original correct ending. Each noun, verb, adjective and adverb was replaced given a user-defined certain probability. The replacement was done with words of the same semantic category taken from the corpus.This augmentation method did not work out very well. We suspect that the models need to see significant changes in the sentence structure to learn some form of statistical discrimination.
We thus tried using another augmentation strategy: a sentence was randomly picked from the context in order to be coherent with the semantic structure of the sentences, and the same kind of word replacement was done.

Given the augmented dataset we then started training all the different models.

## 3 Model

We decided to look at several models: CNN, LSTM, Siamese LSTM and feed forward NN

- CNN followed by LSTM:
  This model takes in input the sentiment related to each sentence of the story. The sentiment analysis feature extraction performed on each sentence gives four decimal outputs in [0,1]: positive, neutral, negative feelings, and the combined result of those. These are given in input to the model. First, the model embeds each value into a vector (dimension = 20). Then, the CNN filters the sentiment vector values of each sentence with window size four. The result is given in input to the LSTM. The output of the LSTM is finally given in input to a dense layer to get the soft-max of the binary decision (right or wrong story).
- Siamese LSTM:
  Siamese networks are networks with two or more identical sub-networks in them. They seem to perform rather well on similarity tasks and have been used to detect sentence semantic similarity, or image difference. Based on LeCun's paper [2], the idea was therefore to use a siamese LSTM architecture with tied weights. As inputs, we had on one side, the story context and on the other, the story ending. Each sentence (represented as a sequence of word indices from the vocabulary, padded to have 43 words in total) first went through a Glove embedding layer, before being passed to the LSTM. Our LSTMs used a 128-dimensional hidden representations $h_t$ and memory cells $c_t$. The cosine distance between the two outputs of the LSTMSs was then computed. The idea was then that the right ending had the smallest cosine distance to the context, compared with the wrong ending.
- Feed-forward Neural Network (FFNN):
  Inspired by Srinivasan's work [5], we used a single feed-forward neural network with a softmax-layer at the end acting as a binary classifier. The models takes in a 9604-dimensional input where we concatenated the combined skip-thought embeddings of the last sentence and the two endings (each with dimension 4800) and the sentiment analysis array (with dimension 4). We used a pre-trained model[6] for skip-thougth embeddings and used its encoder to generate numerical representation of sentences in our datasets. We implemented three-layer networks with ReLu non-linearities, with respectively 3200, 1600 and 800 nodes. In order to prevent overfitting, we regularized with dropout layers with rate 0.2 and used L2 regularization for the output layer.

## 4 Training

- CNN followed by LSTM (**CNN-LSTM**): The binary decision of the network is compared with the expected outcome. The loss function used is the cross entropy loss function, tuned by using the Adam gradient-based optimizer, with learning rate of 0.001.

- **Siamese LSTM**: The training was done on the augmented training dataset, meaning that 2 wrong endings were created for each story. Duplicating a context for each ending, the input size was then 3 times the number of stories of the training dataset. Given that the output of the model was simply a distance between the outputs of the two Siamese LSTM's, the training was based on the mean squared error, optimized with Adam optimizer.

- Feed-forward Neural Network (**FFNN**): the training was based on cross-entropy loss, optimized with Adam optimizer with learning rate of 0.001.

For all models, during training, we created a model checkpoint at every epoch and saved the best model while monitoring the validation accuracy.

## 5 Experiments

We trained our models on both the provided training set and validation set. When training on the training set, we used an enlarged dataset with negative endings as explained previously. On the other hand, when training on the validation set, we split the validation data into two smaller sets, by holding 10% of the data for validation. After training, we evaluated our models on Story Cloze test set.

In the following table, we reported the highest train and validation accuracies reached during training, and the accuracy on Story Cloze test set for the trained models. Unfortunately, training using the siameseLSTM model on a small set of the training sample yielded an validation accuracy of exactly 50%, so we considered the result as being unreliable.

| Model | Train accuracy (%) | Validation accuracy (%) | Cloze Test accuracy (%) |
|---|---|---|---|
| CNN-LSTM | 60 | 55.5 | 46.2 |
| Feed-forward Neural Network | 85.7 | 60.2 | 59.1 |

While training on the validation data, we obtained the following results:

| Model | Train accuracy (%) | Validation accuracy (%) | Cloze Test accuracy (%) |
|---|---|---|---|
| CNN-LSTM | 75 | 55 | 44.5 |
| Feed-forward Neural Network | 88.7 | 73.4 | 68.8 |

From the results above, we can state that the Feed-forward Neural Network model gave the best performance in both the cases. The FFNN model managed to learn story continuation more effectively, because of its better generalization performance. This result might also be related to its feature selection: while we added a context to the ending with sentiment analysis, as stated in Srinivasan's paper [5], selecting only the last sentence from the context seems to provide better results and this might be inherent to human storytelling.

On the other hand, the CNN-LSTM model gave negative results on the Cloze Test, giving positive one for the validation though. The reason behind this could be that the validation set contains similar stories sentiment-wise, while the provided test set is not really well fit for the sentiment analysis.

Overall, by training on the validation set, we obtained better results despite the relatively small samples set. The validation set was provided with the wrong endings, while for the training set we had to create negative endings with data augmentation. Having accurate wrong endings lead to better accuracy in prediction, although since the dataset is small, we were prone to overfitting, which we tried to limit by using regularization techniques.

## 6 Conclusion

The Story Cloze test is a challenging problem which requires some well-studied strategy to be tackled. Forms of feature extraction (e.g sentiment analysis, syntactic analysis, semantic analysis) looked to be the key to obtain significant results. In fact, although some of the input information is inevitably reduced or completely lost during the process, the trained models were able to perform quite meaningful statistical analysis.

## References


[1] T. Miha ylov, A. Frank (2017): Story Cloze Ending Selection Baselines and Data Examination. Heidelberg University.

[2] R. Hadsell, S. Chopra, Y. LeCun (2016): Dimensionality Reduction by Learning an Invariant Mapping. The Courant Institute of Mathematical Sciences, New York University.

[3] Melissa Roemmele, Sosuke Kobayashi, Naoya Inoue, Andrew M. Gordon: An RNN-based Binary Classifier for the Story Cloze Test.

[4] R. Kiros, Y. Zhu, R. Salakhutdinov, et al. (2015): Skip-Thought Vectors

[5] S. Srinivasan, R. Arora, M. Riedl (2018): A Simple and Effective Approach to the Story Cloze Test. Georgia Institute of Technology.

[6] Pre-trained embeddings: https://github.com/ryankiros/skip-thoughts

[7] B. Wang, K. Liu, J. Zhao (2017): Conditional Generative Adversarial Networks for Commonsense Machine Comprehension.

[8] I. Habernal, H. Wachsmuth, I. Gurevych, B. Stein (2018): The Argument Reasoning Comprehension Task: Identification and Reconstruction of Implicit Warrants.

[9] M. Bugert, Y. Puzikov, A. Ruckle, J. Eckle-Kohler, T. Martin, E. Martinez-Camara, D. Sorokin, M. Peyrard I. Gurevych (2017): LSDSem 2017: Exploring Data Generation Methods for the Story Cloze Test.

[10] N. Mostafazaden, N. Chambers, X. He, D. Parikh, D. Batra, L. Vanderwende, P. Kohli, J. Allen (2016): A Corpus and Cloze Evaluation for Deeper Understanding of Commonsense Stories.

[11] N. Mostafazadeh, M. Roth, A. Louis, N. Chambers, J. Allen (2017): LSDSem 2017 Shared Task: The Story Cloze Test.

[12] R. Schwartz, M. Sap, I. Konstas, L. Zilles, Y. Choi N. Smith. (2017): Story Cloze Task: UW NLP System. University of Washington

[13] S. Chaturvedi, H. Peng, D. Roth (2017): Story Comprehension for Predicting What Happens Next. University of Illinois

[14] M. Flor, S. Somasundaran (2017):Sentiment Analysis and Lexical Cohesion for the Story Cloze Task. Educational Testing Service