Coherent Short Story Generation Using Set of Independent Plot Points

Aarsh Prakash Agarwal 150004 Aditi Singh 150048 Pratyush Garg 150521 Shivam Utreja 150682 Shubhanshu Khandelwal 150705 Swapnil Agarwal 150752

***Abstract*—Story generation has always been associated with creative minds and sound writers. However, with advancements in neural networks and learning algorithms, it has now become possible for computers to mimic writing styles and synthesise stories. Our work requires the AI agent to finish stories based on open plot lines supplied to it. The only constraint is that the story should be coherent and readable.**

I. Problem Statement

We aim to generate a logically coherent story through intermediate plot points. We intend to use a possible combination of the two approaches mentioned in literature review as per their merits. The dataset description of each of these approaches are also given in the following sections.
Story Generation is perceived in many different ways today. We have worked on two different approaches as follows-
Approach I
Coherent story generation from independent descriptions, describing a scene or an event.
Approach II
Automatically selecting a sequence of events that meet a set of criteria and can be told as a story; Knowledge-intensive: Rely on a priori defined domain models about fictional worlds, including characters, places, and actions that can be performed.

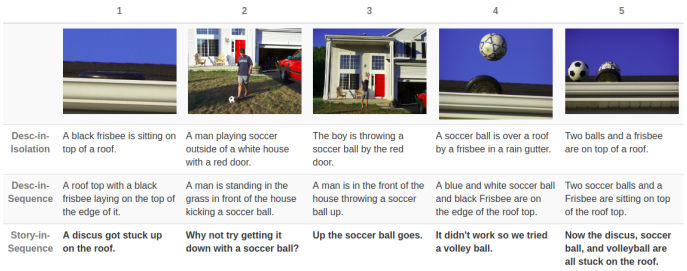
II. Approach I: Phrase Based Statistical Machine Translation (PBSMT)

A. Introduction

In this approach we have tried to cast the problem as **Phrase Based Statistical Machine Translation** problem. For this coherent short stories can be considered as target while the independent short sentences can be considered as source. We are here trying to learn a supervised traslation model that will try to *translate* those independent short sentences into coherent stories.

B. Description of Dataset

We are using **VIST** Visual Story-Telling dataset for our purpose.The dataset includes 81,743 unique photos in 20,211 sequences, aligned both, according to description in isolation(DII) and story in sequence (SIS) formats. The following image highlights the difference between the Description in Isolation(DII) and Story in Sequence(SIS) ways of captioning the images of the same album presented in the same sequence.



	1	2	3	4	5
Desc-in-Isolation	A black frisbee is sitting on top of a roof.	A man playing soccer outside of a white house with a red door.	The boy is throwing a soccer ball by the red door.	A soccer ball is over a roof by a frisbee in a rain gutter.	Two balls and a frisbee are on top of a roof.
Desc-in-Sequence	A roof top with a black frisbee laying on the top of the edge of it.	A man is standing in the grass in front of the house kicking a soccer ball.	A man is in the front of the house throwing a soccer ball up.	A blue and white soccer ball and black Frisbee are on the edge of the roof top.	Two soccer balls and a Frisbee are sitting on top of the roof top.
Story-in-Sequence	**A discus got stuck up on the roof.**	**Why not try getting it down with a soccer ball?**	**Up the soccer ball goes.**	**It didn't work so we tried a volley ball.**	**Now the discus, soccer ball, and volleyball are all stuck on the roof.**

Fig. 1: An example of an album in dataset

Note that the row mentioning Description in sequence is present strictly to highlight the difference between describing a set of images in a sequence and depicting it as a story(SIS). No such description in sequence is available in the dataset. The DII data was generated by showing each image of the same album to a different person(depicted by a different worker ID in the data) and asking them to describe the image. The SIS data on the other hand, was generated by presenting each complete album to a single person in an unordered fashion; asking them to order the images and then describing them as a story in that specific order. The data was preprocessed to convert it into a parallel sentence corpora before using it for phrase-based statistical machine translation. First, the image order used for each album in the SIS dataset was extracted. The DII of the corresponding album were then reordered according to this image order; and then concatenated into one single sentence(including fullstops). The corresponding SIS was also concatenated into a single sentence, in the same image order. These two files; after being tokenized and lower-cased, now acted as our parallel sentence corpora to train, test and tune our machine translation models.

C. Model

1) Description: We have selected Phrase based Statistical Machine Translation **PB-SMT** as our translation model as it has significant advantages over the word based models. It regards *phrases* as the atomic unit instead of *words* which enables it to handle many-to-many translation. It also allows it to use local context in translation. Given the sentences of target language $e_1, e_2, e_3 ..... e_l$ and source language $f_1, f_2, f_3, ... f_l$ which in our case is the *sis* and *dii* respectively, the model can described as:

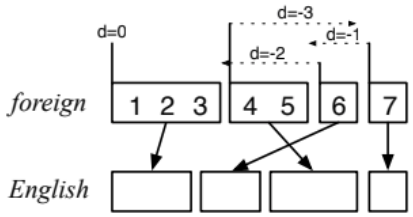
$$e_{best} = argmax_e p(e|f)$$

$$\Rightarrow e_{best} = argmax_e p(e|f) p_{LM}(e)$$

where, $p(e|f)$ is the *translation* model and p_{LM} is the *Language* model. The Translation model can be further broken into *phrase transition model* ϕ and *reordering* model d.

$$p(f|e) = \Pi_{i=1}^{l} \phi(f_i|e_i) d(start_i - end_{i-1} - 1)$$

2) Distance based Reordering: We are using distance based reordering for our model d.



phrase	translates	movement	distance
1	1–3	start at beginning	0
2	6	skip over 4–5	+2
3	4–5	move back over 4–6	-3
4	7	skip over 6	+1

Fig. 2: Distance based Reordering

The distance score is calculated as α^d where $0 < \alpha < 1$. The intuition behind using this type of system is that you get higher reordering score if translation of phrases happens in the ordered form, thus resulting in higher translation probability.

3) Learning Phrase translation table: In order to learn the phrase translation table we have perform three steps:

- Word alignment: This is done using models like IBM model, HMM model.

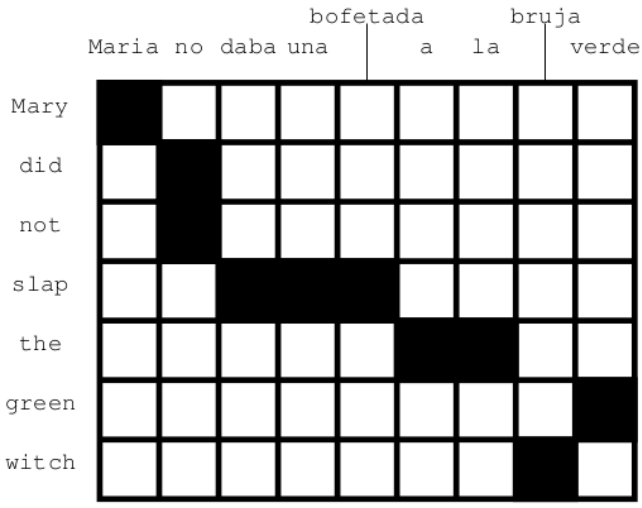


Fig. 3: Word alignment in of target and source sentence

- Phrase Extraction: phrases are recursively extracted from word alignment matrix by recombining aligned words. Large phrases can be constructed by recombining smaller phrases. The details can inferred from Figure 4 and Figure 5.

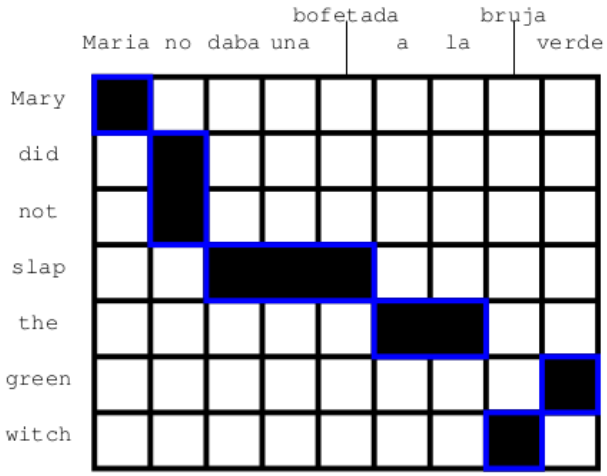


Fig. 4: Phrases constructed by recombining aligned words

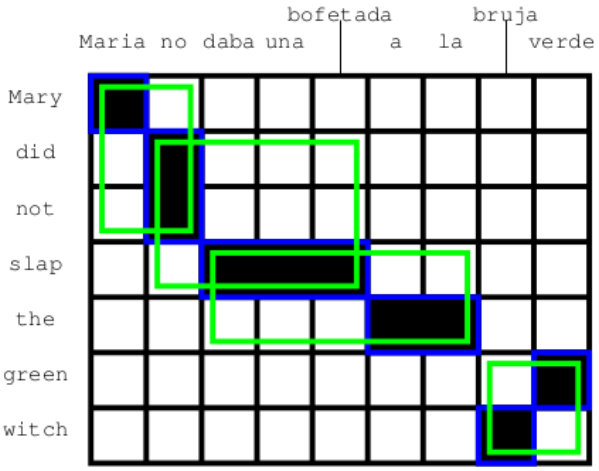


Fig. 5: Larger phrases constructed from previous phrases

- Scoring Phrase Translation: Extracted phrases can be given scores as

$$\phi(e|f) = \frac{count(e, f)}{\sum_{f_i} count(e, f_i)}$$

which is essentially number of sentences in which phrase is present divided by the total number of sentences

4) Weighted Model: The weighted model can be developed with weights λ_ϕ, λ_d and λ_{LM} for Translation model, reordering model and language model respectively.

$$e_{best} = argmax_e \Pi_{i=1}^{l} \phi(f|e)^{\lambda_\phi} d(start_i - end_{i-1} - 1)^{\lambda_d}$$
$$\Pi_{i=1}^{|e|} p_{LM}(e_i|e_1, e_2....e_{i-1})^{\lambda_{LM}}$$

The model can be log linearized for maximization.

5) EM Algorithm: The model can be learned using popular Expectation Maximization Algorithm which is as follows

- Initialization: Uniform model, all $\phi(e, f)$ are equal.
- Expectation: Estimate likelihood of all possible phrase alignments for all sentence pairs.
- Maximization:
  - Collect counts for phrase pairs (e, f) weighted by alignment probability.
  - Update phrase translation probabilities $p(e, f)$.

D. Implementation

We have used two different toolkits namely **THOT** and **MOSES** for our implementation. Two models were trained on each one of them. Details are as follows:

- MOSES:
  - 5 gram vector was used as Language Model using KenLM library.
  - Word Alignment was done using GIZA++ while the tuning was done using MERT.
- THOT:
  - 3 gram vector was used as Language Model.
  - Word Alignment was done using HMM.

Both the models were trained on 29,000 training sentences and tuned on 1,000 validation set. Models were tested on data set of 100 sentences.

III. Approach II: Story Generation with Plot Graphs

A. Introduction

The approach is motivated by the intuition that to tell a story one must first have a theme in mind, subsequently the template of the story is obtained and finally the entire story emerges. Using combinations of pre-written crowd-sourced stories on different themes, we attempt to generate new legal stories around similar templates that make sense and are coherent. We generate plot graphs to model the temporal relations and other fictional interactions between different events and characters. A traversal of this graph gives us the template after which we can randomly sample sentences for each point in the template and return the story.

B. Description of the Dataset

The dataset we used was crowd-sourced by the author of the paper II. It was obtained after a survey in which the participants were asked to write simple stories on alloted themes using the characters "John","Sally" and "Amy". The topics in the corpus of narrative examples are "robbery", "affairs", "airport", "coffee", "gas", "new-movie", "pharmacy", "proposal" and "restaurant". They were asked not to use pronouns and to keep the story short (8-10 lines). Even individual lines were minimalistic and only served to convey an action being taken.

All the stories on a particular theme were concatenated together in the form of a file with the extension ".story". We were also provided with a clustering of related sentences from different stories. These are classified as events and were clubbed together in a file with the extension ".gold". A representative image of the two files is given.

```
@ John sees Sally
2 John watched Sally.
88 John noticed Sally, the bank teller.
141 John saw Sally behind the second teller window with a customer.
241 John recognized Sally.
305 John saw Sally standing behind the counter.
###
@ John scans the bank
17 John looked around the bank.
240 John scanned the bank lobby.
582 John scanned the inside of the bank for cameras.
662 John looked to see which teller was available.
```

Fig. 6: A .gold file

```
0 John walked into the bank.
1 John sat down.
2 John watched Sally.
3 John walked over to Sally.
4 John pulled out a gun.
5 John asked Sally to give her all of the money.
6 Sally screamed.
7 Sally tried to hit the panic button.
8 Sally went to the safe.
9 Sally got all the money from the safe.
10 John gave Sally a bag for the money.
11 Sally put the money in the bag.
12 John ran out of the bank with the money.
13 Sally saw the cops coming.
14 Sally let the cops in and told them what happened.
```

Fig. 7: A sample story

C. Generation of the Plot-Graph

The main problem to be solved is the generation of plot graphs. A plot graph defines the space of legal story progression and ultimately determines possible events at any given point in time. The graph is a tuple $G = (E, P, M)$ where E is the set of all possible plot points and $P \subset \{x \rightarrow y | x, y \in E\}$ is a set of ordered pair of events that describe precedence constraints and $M \subset \{(x, y) | x, y \in E\}$ is a set of mutual exclusion relations.

Learning the plot graph comprises of three stages. The first stage, a corpus of narrative examples is crowd-sourced and clustered which we already have. In the second stage precedence between different events are learned. Finally, in the third stage, mutual exclusion between events are learned and optional and conditional events are defined.

1) Learning of Precedence Constraints: Given two events e_i and e_j, the hypothesis representing the two possible orderings $e_i \rightarrow e_j$ and $e_i \rightarrow e_j$ is calculated based on a binomial distribution. If the hypotheses are above a confidence level $T_p (= 0.4)$ the corresponding ordering is added to the set

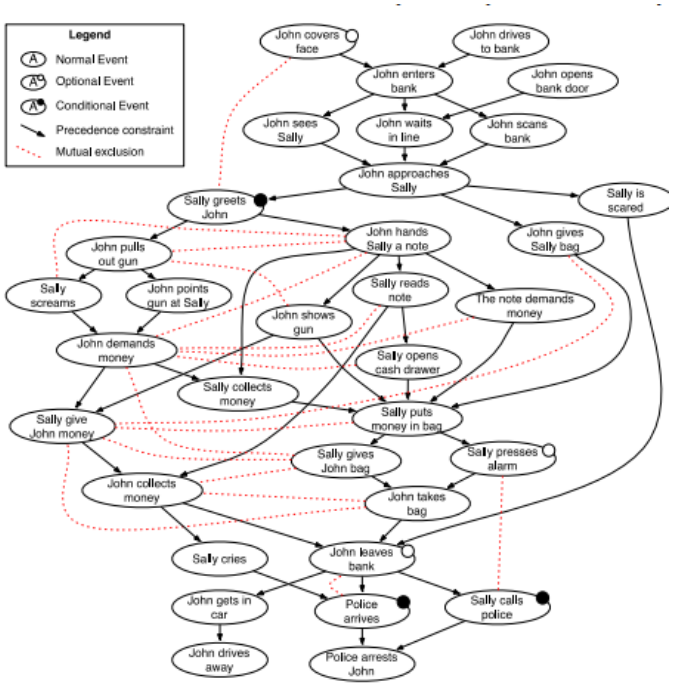


Fig. 8: Plot graph for 'Robbery' theme

P. It is possible to reject both the orderings. If a precedence is established, it is assumed that the preceeding event is necessary for the other to occur.

2) Learning Mutually Exclusive and Optional Events: The mutual exclusion of two events is a parameter of their inter-dependence which is modelled using the Mutual Information between them.

$$MI(E_i, E_j) = C(0,0) + C(0,1) + C(1,0) + C(1,1)$$

where

$$C(a,b) = p(E_i = a, E_j = b) * log \frac{p(E_i = a, E_j = b)}{p(E_i = a)p(E_j = b)}$$

and $p(E_i = 1)$ indicates the probability that the event E_i occurs and $p(E_i = 0)$ indicates the probability that it does not come to pass. If the mutual information is greater than a threshold value, they are considered mutually exclusive. If events e_i and e_j are mutually exclusive then they cannot occur together in a story. However if there is also an ordering relation between them then the necessity condition is abandoned and the preceeding event is considered optional given that it is not mutually exclusive with other preceeding events.

The system generates stories by stochastically adding events to the story such that no precedence constraints or mutual exclusion relations are violated. The story ends when there are no legal events left to be added.

The psuedocode is given in Fig 9.

Function GENERATESTORY (*plot-graph*)
  For each optional event $e \in$ *plot-graph* **do**
    Insert a link between each direct predecessor of *e* and each direct successor of *e*
  Let *story* $\leftarrow \emptyset$, *best* $\leftarrow -\infty$
  Repeat *n* **times**
    new-story $\leftarrow$ WALKGRAPH(*plot-graph*)
    value $\leftarrow$ EVALUATEFITNESS(*new-story*)
    If *value* > *best* **then do**
      story $\leftarrow$ *new-story*, *best* $\leftarrow$ *value*
  Return *story*

Function WALKGRAPH (*plot-graph*)
  Let *story* $\leftarrow \emptyset$
  While not ISCOMPLETESTORY(*story*, *plot-graph*) **do**
    Let *options* $\leftarrow$ EXECUTABLEEVENTS(*plot-graph*, *story*)
    Let *e* $\leftarrow$ pick an event from *options*
    story $\leftarrow$ *story* + *e*
    plot-graph $\leftarrow$ UPDATEGRAPH(*plot-graph*, *story*)
  Return *story*

Fig. 9: Pseudocode for Story Generation using Plot-Graphs

IV. RESULTS

A. Approach I

The model trained with MOSES on test data resulted in a BLEU score of 0.117829 while the one trained with THOT resulted in BLEU score of 0.271018. Also, the sentences generated from THOT were more readable and meaningful although these were inconsistent with punctuations like full-stop. While the output generated by both these models weren't found completely coherent, they can be of quite much use to a writer to work upon.

The reasons of lack of efficacy can arise because of lack of training data, we had only 32000 images. The tuning of the model, needless to say, must also be paid attention to. The training itself took 10-12 hours to complete, while the tuning ran for another 14-18 hours on mere 1000 sentences been used. An example can be seen in TABLE: 1

B. Approach II

The model output, as can be seen below, was quite coherent. This model can prove quite helpful but only for particular themes. Table II shows some examples generated on the specified themes. This method is logically simple and intuitive. However, the possible outcomes are limited by the number of stories. Our main issue with this approach was the dataset which was incoherent in itself and inconsistent at best. The clustering is poorly done and there are no clear demarcations. For some events, the set is so small that no precedence or exclusion relations can be learned and we can glean no information at all. These are the events which appear randomly in the output.

Description in Isolation	Thot output	Moses output
he writing to open christmas gifts is quite sloppy . a cardboard box is filled with an invitation card and random gifts . a cartoon character that is sitting on the table . a clear plastic bag of googly eyes sits on a table . someone has left a note on a loaf of bread , saying it is very delicious .	watching over me . a sweet treat to celebrate all that life has to never forget that their are people watching me and card . and a box filled with gifts galore . a doll to keep me company in case i get lonely . a . . . galore . a doll to keep me company in case i get lonely . a bag of googly eyes the over me . a sweet treat to celebrate all that life has to offer arrived it all started with a simple .	To sweep theresa manly manner . hailey simplified works depict the virgin mary en er pocket and chatted with a simple card . as you can see donald gifts for the sewing bag of googly a monster . she is even more beautiful than a delicious treat ! bright cheerful . before the night was over some super uncoordinated treat

TABLE I: Examples of Thot and Moses implementation on above example

THEME : ROBBERY	THEME : COFFEE	THEME : GAS
1) John scanned the inside of the bank for cameras. 2) John recognized Sally. 3) John took a note out of his pocket. 4) Sally looked frightened. 5) John drove away with the money. 6) John handed Sally a bag. 7) Sally gave him the money. 8) John picked up the money.	1) John opened the cabinet. 2) John turned the coffee pot on. 3) John poured hot water into the mug. 4) John added sugar and cream. 5) John scooped out some coffee. 6) John poured himself a cup of coffee. 7) John drank his mug of coffee.	1) John drove into the gas station parking lot. 2) John drove to the third gas pump. 3) John held on to the handle. 4) John went up to the pump display. 5) John put the nozzle into his car. 6) John pressed the button labeled for eighty-seven octane fuel. 7) John started his car. 8) John returned to his car. 9) John drove away.

TABLE II: Examples of plot graph generated outputs

V. Future Scope

This problem was interesting and clear solutions as of yet are not there in literature. With this project we attempted to bring some of these approaches to light. IN this process we were exposed to the many drawbacks present in the current approaches. Instead of direct translation, we can explore a two step process to extract context more properly in approach I while the crowd-sourcing of better datasets may provide better results in approach II.

VI. Contribution

150705	Shubhanshu Khandelwal	18
150752	Swapnil Agarwal	15
150686	Shivan Utreja	18
150048	Aditi Singh	10
150004	Aarsh Agarwal	20
150521	Pratyush Garg	19

VII. Acknowledgement

We thank Prof. Harish Karnick for giving us this opportunity to explore story synthesis. It was a challenging topic and would not have been possible without his insightful lectures and course-material. This project was an incredible learning experience for all of us and hopefully taught us how to design, plan and create better.

We also gratefully acknowledge the support of Dr. Boyang Albert Li who very graciously provided us the dataset for his paper upon request.

VIII. References

[I] Li, B., Lee-Urban, S., Appling, D.S., and Riedl, M.O. 2012. *Crowdsourcing Narrative Intelligence. Advances in Cognitive Systems*, 2, 25-42. https://pdfs.semanticscholar.org/8bea/9293fc542eb5836929da4457e58135794904.pdf

[II] Li B., Lee-Urban S., Johnston G., and Riedl, M.O. *Story Generation with Crowdsourced Plot Graphs* https://www.cc.gatech.edu/~riedl/pubs/aaai13.pdf

[III] Chambers, N. and Jurafsky, D. 2008. *Unsupervised Learning of Narrative Event Chains.* Proc. of the 46th Annual Meeting of the Association for Computational Linguistics. https://www.aclweb.org/anthology/P/P08/P08-1090.pdf

[IV] Jain, Parag and Agrawal, Priyanka and Mishra, Abhijit and Sukhwani, Mohak and Laha, Anirban and Sankaranarayanan, Karthik. *Story Generation from Sequence of Independent Short Descriptions*. https://arxiv.org/pdf/1707.05501.pdf

[V] Philipp Koehn, Hieu Hoang, et al. 2007. *Moses: Open source toolkit for statistical machine translation*. http://www.aclweb.org/anthology/P07-2045 In Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions.

[VI] @InProceedingsOrtiz2014, author = Daniel Ortiz-Martínez and Francisco Casacuberta, title = The New Thot Toolkit for Fully Automatic and Interactive Statistical Machine Translation, booktitle = Proc. of the European Association for Computational Linguistics (EACL): System Demonstrations, year = 2014, month = April, address = Gothenburg, Sweden, pages = "45–48",

[VII] PB-SMT slides by Kohen http://statmt.org/book/slides/05-phrase-based-models.pdf

[VIII] Visual Storytelling Dataset (VIST) - http://visionandlanguage.net/VIST/dataset.html http://boyangli.co/data/openni.zip