Understanding Connected Text

John was hungry. He went into Goldstein's and ordered a pastrami sandwich. It was served to him quickly. He left the waitress a large tip.

- What is Goldstein's?
- What did John eat?
- Who made the sandwich?
- Who took John's order?
- Who served the sandwich?
- Why did John leave a large tip?

Scripts

- A **script** is a stereotypical sequence of events in a standard situation.
- Scripts organize the knowledge associated with common events.
- Scripts represent a causal chain of events and states that describe a situation.
- People usually omit many of the parts of the causal chain when telling a story. (Indeed, a person who *doesn't* leave out much is considered to be boring!)
- Scripts are essential to story understanding as a means of filling in the details that are not explicitly mentioned but would normally be inferred.

A simple restaurant script

$RESTAURANT

1. go to a restaurant
2. be seated
3. read menu
4. order food
5. eat food
6. pay for meal
7. leave the restaurant

Each event in a script is called a **scene.** Each scene may have its own set of stereotypical subevents.

Understanding a story

John was hungry. He went into Goldstein's and ordered a pastrami sandwich. It was served to him quickly. He left the waitress a large tip.

1. go to a restaurant	-> John went to Goldstein's.
2. be seated	-> John sat down.
3. read menu	-> John read a menu.
4. order food	-> John ordered a pastrami sandwich.
5. eat food	-> John ate the pastrami sandwich.
6. pay for meal	-> John paid for the meal and left a tip.
7. leave the restaurant	-> John left Goldstein's.

Script Tracks

A script may have several different tracks that represent variations of the situation.

For example, the restaurant script may have different tracks for:

- Cafeteria: seat yourself, no waiter or waitress, no tipping
- Fast food: seat yourself, no waiter or waitress, no tipping, paying occurs after ordering but before eating, eating may occur inside or outside the restaurant
- Fancy restaurant: may be several waiters/waitresses, wine list, more courses

A Subway Script - in Detail

$SUBWAY

1. enter subway station
2. go to turnstile
3. put token in turnstile and go through
4. go to platform
5. wait until train comes
6. enter train and find seat
7. leave train when destination reached

Script Roles

Each script has a cast of characters specified by **roles.**

For example, $SUBWAY contains the roles:

&PATGRP	group of subway riders
&CASHIER	cashier
&CONDUCTOR	conductor
&DRIVER	person driving the train
&SUBORG	the subway organization

Props

Each script has a set of associated objects called **props.**
For example, $SUBWAY contains the props:

&TOKEN	subway token
&FARE	money paid for a token
&TURNSTILE	turnstile
&PLATSEAT	seat on a platform
&SUBWAY	the train
&SUBWAYCAR	one of the cars of the train
&CARSEAT	a seat on a car
&STRAP	a strap for a patron to grasp
&EXITGATE	the gate from the platform leading to the destination.

Settings

Each script has a list of places where the events take place which are called **settings.**

For example, the $SUBWAY has three main settings:

the origin station
the inside of the train car
the destination station

Together, the roles, props, and settings make up the **script variables.** The script variables are then instantiated during script applcation by information found in the story, or by default values.

Determining what script should be active can be tricky ...

A classic mistake (by FRUMP, not SAM).
Given the headline:

POPE'S DEATH SHAKES THE WESTERN HEMISPHERE.

The system produced the following summary:

THERE WAS AN EARTHQUAKE IN THE WESTERN HEMISPHERE.

THE POPE DIED.

Event representations

- Each sentence is processed by ELI which produces a CD representation for the event.
- The CD representation allows many different sentences to be mapped to the same concept, reducing the size of the script.
- The script inferences are based on conceptual primitives, so they don't have to be duplicated.
- Script roles are filled by PPs (picture producers) that belong to the #animate class.

The patron role &PATGRP can be filled by a PP of the class #person (e.g., "John") or #group (e.g., "a group from NY").

The subway organization role &SUBORG can be filled by a PP of the class #organization.

Another Story

Friday evening a car swerved off Route 69. The vehicle struck a tree. The passenger, a New Jersey man, was killed. David Hall, 27, was pronounced dead at the scene by Dr. Dana Blauchard, medical examiner. Frank Miller, 32, of 593 Foxon Rd., the driver, was taken to Milford Hospital by Flanagan Ambulance. He was treated and released. No charges were made. Patrolman Robert Onofrio investigated the accident.

Q1: Was anyone killed?
A1: Yes, David Hall died.

Q2: Was anyone hurt?
A2: Yes, Frank Miller was slightly injured.

Q3: Why was Miller hurt?
A3: Because the automobile hit the tree.

Q4: Did Miller go to the hospital?
A4: Yes, he was in the Milford hospital.

Q5: How did Miller get to the hospital?
A5: The Flanagan Ambulance Company took him to the Milford hospital.

SAM in Action

John went to a restaurant. He ordered a hot dog. The waiter said they didn't have any. He asked for a hamburger. When the hamburger came it was burnt. He left the restaurant.

Q1: Did John sit down at the restaurant?
A1: Probably.

Q2: Did John order a hot dog?
A2: Yes.

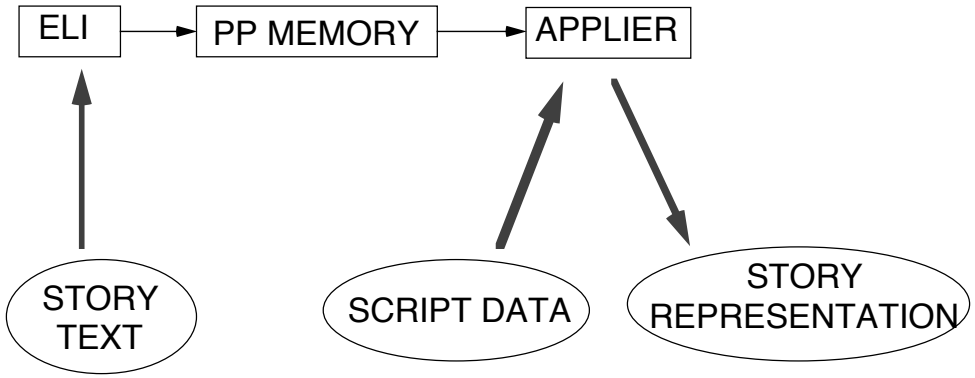
Q3: Did John eat a hot dog?
A3: No, the waiter told John the management was unable to give it to him.

Q4: What did the waiter serve John?
A4: The waiter served John a hamburger.

Q5: Why didn't John eat the hamburger?
A5: Because the hamburger was overdone.

Q6: Did John pay the check?
A6: No, John was angry because the hamburger was overdone so he left the restaurant.

SAM: Script Applier Mechanism



Using scripts

- **Script Instantiation**: Deciding which script is being referred to in a story and bringing it into the system.

  1. The *Pattern Matcher* determines whether the input conceptualization matches any of the key patterns in a script.

- **Script Application**: Applying the script to construct a causal chain and fill in important details. The script can fill in gaps or identify the relationships between items.

  2. The *Predictor* sets up predictions that are likely to follow.
  3. The *Instantiator* merges the current input into the script.

An example of the script application process

Sunday morning Enver Hoxha, the Premier of Albania, and Mrs. Hoxha arrived in Peking at the invitation of Communist China. The Albanian party was welcomed at Peking Airport by Foreign Minister Huang. Chairman Hua and Mr. Hoxha discussed economic relations between China and Albania for three hours.

ELI processes the first sentence and produces a CD representation. The CD form looks something like this:

PTRANS: actor = #GROUP (Enver and Mrs. Hoxha ...)
object = #GROUP (Enver and Mrs. Hoxha ...)
to = (INSIDE (LOC (PEKING)))
time = (Sunday morning)

↕

$INVITATION: INVITER = (Communist China)
INVITEE = nil
INVITOBJ = nil

The Tokenizer

References to PPs are replaced by tokens, and the information associated with "permanent tokens" (well-known people and places) is retrieved.

GROUP0 = #GROUP (Enver and Mrs. Hoxha ...)

HUM0 = #PERSON (Mrs. Hoxha ...)

HUM1 = #PERSON (Enver Hoxha, Premier of Albania, ...)

POLIT0 = #POLITY (Peking, China)

POLIT1 = #POLITY (Communist China)

POLIT2 = #POLITY (Albania)

The Script Applier

- The script applier searches its scripts and finds that an invitation is part of the $VIPVISIT script. $VIPVISIT is activated, and China is assigned to the script role &INVITER.
- Now SAM expects the context to describe events related to state visits. SAM checks to see whether the arrival (PTRANS) fits into the $VIPVISIT script.
- One of the script's predictions is that the invited person will travel to the destination of the invitation. So SAM integrates the PTRANS into the $VIPVISIT script and infers that Enver and Mrs. Hoxha are the invitees.

Sentence #2

The Albanian party was welcomed at Peking Airport by Foreign Minister Huang.

ELI parses this directly into a $VIPWELCOME event since the $VIPVISIT script is active.

$VIPVISIT: WELCOMER = #PERSON (Foreign Minister Huang)
WELCOMEE = #GROUP (Albanian Party)

(LOC (PROX (PEKING AIRPORT)))

$VIPVISIT predicts that the invitee will arrive at the destination. SAM therefore infers that the Albanian Party refers to Mr. & Mrs. Hoxha. The word "airport" also allows SAM to infer that they came by plane.

Sentence #3

Chairman Hua and Mr. Hoxha discussed economic relations between China and Albania for three hours.

ELI produces the following conceptualization:

Chairman Hua and Mr. Hoxha  MTRANS:
MOBJECT = #CONTRACT
TYPE = economy
PARTY = #GROUP (Albania, China)
INST = Chairman Hua and Mr. Hoxha

SPEAK

$VIPVISIT predicts that there should be an object of the visit, so SAM has been looking for one. The contract is therefore bound to the script variable &INVITOBJ.

The summary

The final story representation has recognized three episodes in the $VIPVISIT script:

1. an episode in which a VIP travels to Peking.
2. a official-greeting episode at the airport.
3. an official-talks episode.

SAM then produces the following summary:

Premier Enver Hoxha, the Albania government head, and Chairman Hua Kuo-Feng, the China government head, discussed Albania/Communist-China economic affairs in Peking, China two days ago.

Question-answering

SAM can answer some non-trival questions about the story:

Q1: Who went to China?
A1: Premier Enver Hoxha and Mrs. Hoxha

Q2: How did they get to China?
A2: Mrs. Hoxha and Premier Enver Hoxha flew to Communist China.
Note that "they" is easily recognized as Mr. & Mrs. Hoxha. "How did they get ..." is answered by searching for the instrumental role.

Q3: Why did Enver Hoxha go to China?
A3: Because Chairman Hua Kuo-Feng and Mr. Enver Hoxha wanted to discuss China/Albania economic affairs.

PAM: Plan Applier Mechanism (Wilensky 1978)

Many stories require knowledge about plans and goals in order to understand how one sentence relates to another.

Examples:

John needed money for a down payment on a house.
He called his sister.

John wanted to become a foreman.
He went to get some arsenic.

John was hungry.
He took out the yellow pages.
He took out some ground beef.
He took out Popular Mechanics (?)

PAM in Action

John wanted money. He got a gun and walked into a liquor store. He told the owner he wanted some money. The owner gave John the money and John left.

Q1: Why did John get a gun?
A1: Because John wanted to rob the liquor store.

Q2: Why did John threaten the shopkeeper?
A2: Because John needed to get some money.

Q3: Why did the shopkeeper give John the money?
A3: Because the shopkeeper didn't want to get hurt.

Q4: Why did John leave?
A4: Because John didn't want to get caught.

Different views

Story from John's point of view:

I needed to get some dough. So I got myself this gun, and I walked down to the liquor store. I told the shopkeeper that if he didn't let me have the money then I would shoot him. So he handed it over. Then I left.

Story from the owner's point of view:

I was minding the store when a man entered. He threatened me with a gun and demanded all the cash receipts. Well, I didn't want to get hurt, so I gave him the money. Then he escaped.