CS 626: NLP, Speech & the Web

Computational Humour

Devendra Singh Chaplot (100050033)
Nishanth Dikkala (100050052)
Vipul Singh (100050057)

Introduction

Q) Why can’t a motorcycle run a marathon?

Introduction

Q) Why can’t a motorcycle run a marathon?

A) Because it is two(too) tired

Introduction

Q) Why can't a motorcycle run a marathon?

A) Because it is two(too) tired

Will a computer understand this?

Introduction

Computational Humour
The branch of computer linguistics and AI that uses computers in Humour Research

It is a relatively new area in AI with its first dedicated conference starting in 1996

Contents

1. Motivation
2. Theories of Humour
3. Humour Recognition
4. Humour Generation
5. Computer Model of ‘Sense of Humor’
6. Conclusion

"Humour is a very interesting way to look at artificial intelligence because at some point something has to have two meanings, which is not easy to do with a computer." – Jason Rutter, Manchester University.

Motivation

Humour is **AI Complete**

A problem is AI-complete when difficulty of solving it is equivalent to that of solving the central artificial intelligence problem – making computers as intelligent as people

Motivation

Future of **Human Computer Interaction**

Humour forms one of the integral part of everyday interaction between human beings

Computers should be able to produce and react to humor as we do

Motivation

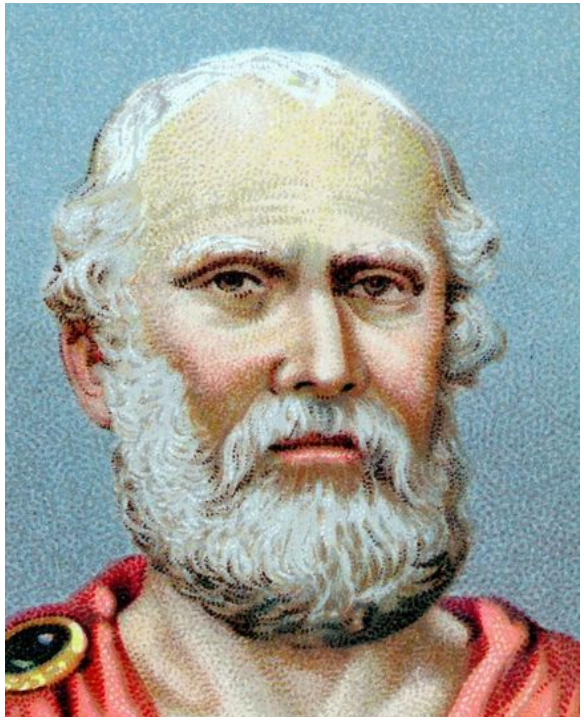
Complex and Puzzling human behaviour

Good quality humor requires decent understanding of situations and normally it is the privilege of talented individuals

THEORIES OF HUMOUR

Superiority Theory

- Traces back to Plato and Aristotle
- A person laughs about the misfortunes of others
- A person asserts his superiority on the background of shortcomings of others



Example:

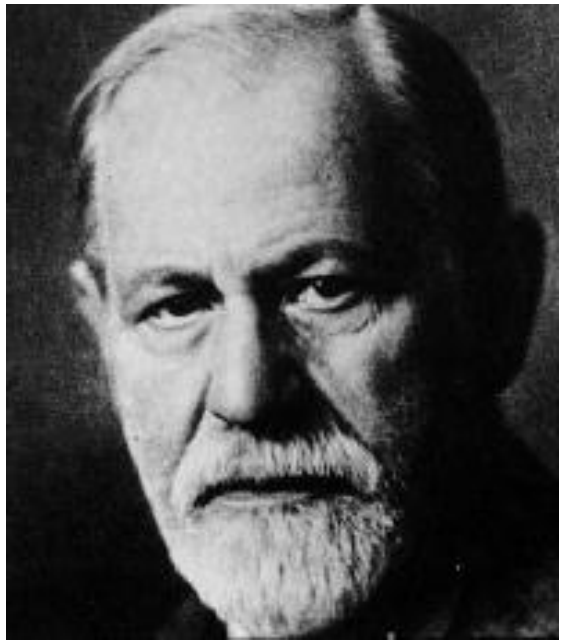
A man spoke frantically into the phone, "My wife is pregnant and her contractions are only two minutes apart!"

"Is this her first child?" the doctor asked.

"No, you idiot!" the man shouted. "This is her husband!"

Relief Theory

- Sigmund Freud and Herbert Spencer prominent theorists
- Joke about things which make people unsure/uncomfortable
- Jokes about religion, politics, sex, ethnic differences
- Release of tension and psychic energy

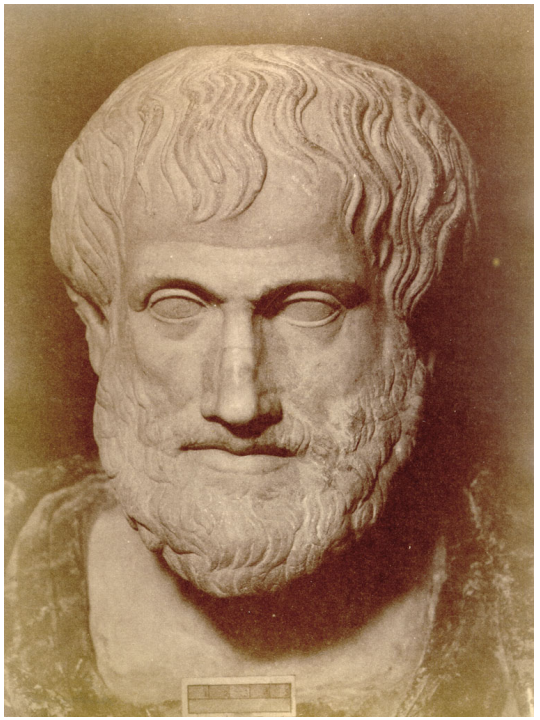


Example:

Women are like programs and men are like programmers, no matter what happens, it is always the mistake of the programmer.

Incongruity Theory

- A leading approach, has its origin in comments made by Aristotle in *Rhetoric*
- The joke has two parts : setup & punch line
- Joke generation due to disagreement in parts
- Main point is not incongruity but realization and resolution



Example:

I said to the Gym instructor: “Can you teach me to do the splits?”

He said: “How flexible are you?”

I said: “I can't make Tuesdays and Fridays.”

Incongruity Theory

Incongruity arises from ambiguity

Humour and lexical ambiguity:
A car owner after coming back from a party finds the sticker "parking fine" on his car. He goes and thanks the policeman for appreciating his parking skill.

Humour and structural ambiguity:
Teacher: What do you think is the capital of Morocco?
Student: What do you think?
Teacher (Angrily): I do not think, I know.
Student: I … do not think I know.

Humour Recognition

Tragedy + Time = Humour

Humour Recognition

- Humour specific stylistic features
  - **Antonymy**: Incongruity, opposition, apparent contradiction
    - Example: Always try to be modest and be proud of it!
    - Identified using WordNet and similar-to antonymy relation among parts of speech
  - **Adult Slang**: Based on adult slang, popular
    - Example: Artificial Insemination: procreation without recreation
    - Extract from WordNet Domains, all the synsets labelled with domain SEXUALITY

Humour Recognition

- Content based learning: general text classification problem
  - **Naive Bayes**: estimate category of a document using joint probabilities of words and documents
  - **Support Vector Machines**: binary classifiers which find hyper-plane that best separates a set of +ve examples from a set of -ve ones.

Classifier	One-liners Reuters	One-liners BNC	One-liners Proverbs
Naïve Bayes	96.67%	73.22%	84.81%
SVM	96.09%	77.51%	84.48%

Humor-recognition accuracy using Naïve Bayes and SVM text classifiers.

Heuristic	One-liners Reuters	One-liners BNC	One-liners Proverbs
Alliteration	74.31%	59.34%	53.30%
Antonymy	55.65%	51.40%	50.51%
Adult slang	52.74%	52.39%	50.74%
ALL	76.73%	60.63%	53.71%

Humor-recognition accuracy using alliteration, antonymy, and adult slang.

Humour Generation

Tragedy + Time = Humour

Humour Generation

We focus on verbal humor generation. At the heart of verbal humor are natural language phenomenon as shown in the following examples.

- Multiple word senses
  Parking 'fine'. Monetary fine or compliment?

- Antonymy
  Always try to be modest and be proud of it

- Homography (words spelt the same way but meaning different things)
  "pen" for writing instrument or animal enclosure

- Structural Ambiguity
  I do not think, I know vs I do not think I know

and many more

Humorous riddle generation systems

- JAPE – A research prototype (worked as a proof of concept)
  JAPE – Joke Analysis and Production Engine

- STANDUP – A practical large-scale working system. Tested on children with positive results. Builds on the concepts of JAPE with some changes.
  STANDUP – System To Augment Non-speakers Dialogue Using Puns

JAPE

- Graeme Ritchie and Kim Binsted – 1994
- Designed to generate question-answer-type puns from a general lexicon.

A deeper look into JAPE's architecture

- Symbolic Rule: An abstract rule with variables not yet associated hard values.
  Some examples:
  - $x^2 - y^2 = (x-y)(x+y)$ and other algebriac rules.
  - Prolog rules e.g child(X,Y).

- JAPE uses three types of symbolic rules
  - Schemas
  - description rules
  - templates

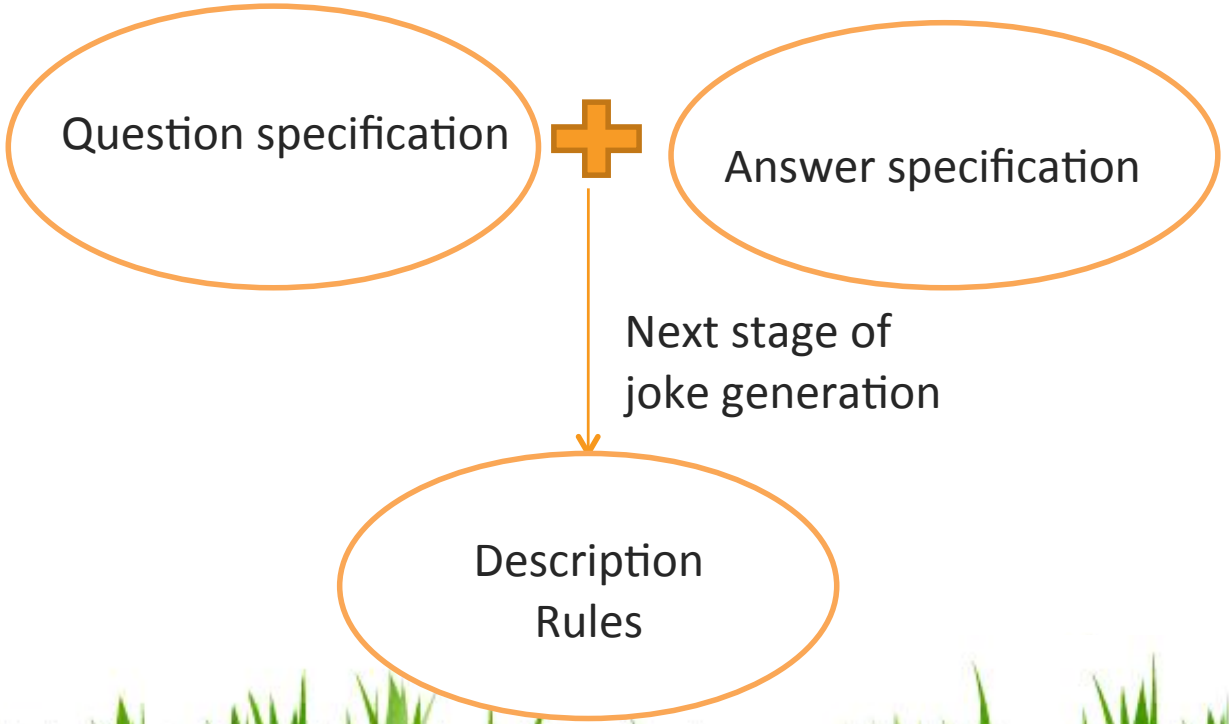
  Characterizes the possible linguistic structures the pun can take using the symbolic rules

Working of JAPE - Schemas

- **Schema:** Consists of 5 parts. Each containing prolog like rules.
- An example schema:
  Header*:* newelan2(NP, A, B, HomB)
  Lexical preconditions: nouncompound(NP,A,B), homophone(B,HomB), noun(HomB)
  Question specification: {shareproperties(NP, HomB)}
  Answer specification: {phrase(A,HomB)}
  Keywords: [NP, HomB]
- Relations and properties are expressed in **Prolog-style** (logic-like) notation, with predicates applied to arguments.

Lexical preconditions

- Constraints in the **lexical preconditions** category can involve
  - syntactic categorisation (e.g. a lexeme is a noun),
  - phonetic relations (e.g. two items rhyme),
  - structural relations (e.g. an item X is a compound noun made up of components Y and Z), and
  - semantic relations (e.g. one lexeme is a hypernym of another).



Working of JAPE - Schemas

Header: newelan2(NP, A, B, HomB)

Symbolic name

Parameters of the schema (variables)

Lexical preconditions:
nouncompound(NP,A,B),
homophone(B,HomB), noun(HomB)

NP should be a noun compound of A and B (noun phrase)

A – Computer

B – Screen

HomB should be a noun

B and HomB should be homophones (similar sounding words). Implemented by restricting B and HomB to lie within the current threshold for phonetic similarity

B – Screen

HomB - Scream

Description Rules

Encode possible linguistic variations,
given core values from the schema instantiation.

Question specification:{shareproperties(NP, HomB)}

Answer specification:
{phrase(A,HomB)}

Header: shareproperties(X,Y)
Preconditions: meronym(X, MerX), synonym(Y, SynY)
Template specifier: [merHyp, MerX, SynY]

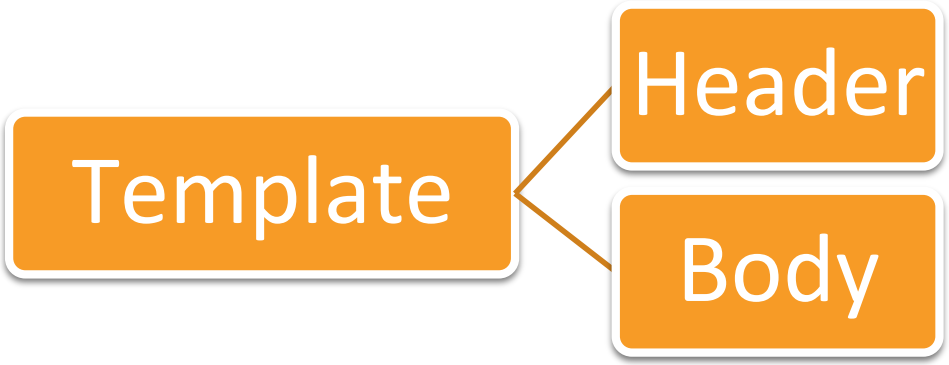
X - Computer Screen
Y – Scream
MerX – pixels
SynY – cry

A sample description rule.
Preconditions specify further lexical properties and relations

Template specifier passed on to the next stage of the system

Working of JAPE - Templates

- Template is a fixed string of text with some blank slots for inserting textual material.



Templates

- Header is matched with the expressions provided by the description rules and the corresponding variables instantiated.

- Body is a skeletal textual structure
  E.g What do you call a NP(X,Y).

- NP(cry,pixels) recursively matches to set of valid phrase templates.

  NP(cry,pixels) ⟶ NP(cry) that has NP(pixels)

"Analyzing humor is like dissecting a frog. Few people are interested and the frog dies in the process." – E.B. White

Finally!! we have generated a joke! Phew.

Q. What do you call a cry that has pixels?

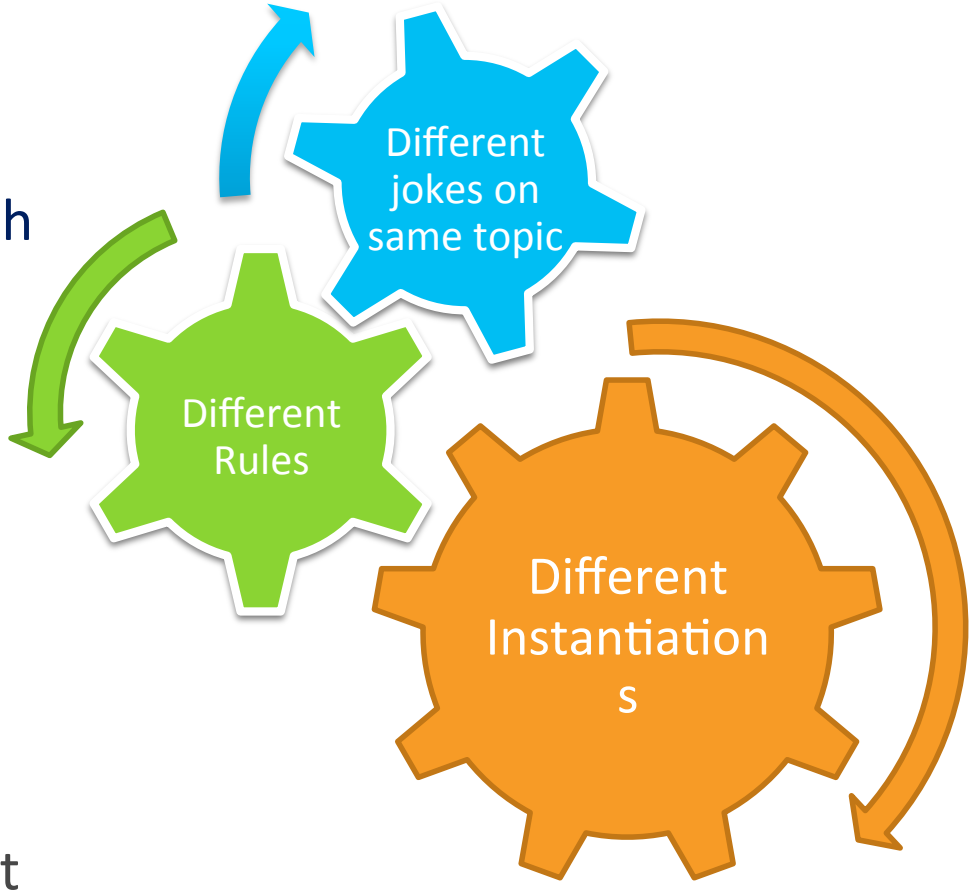
A. A computer scream

Different jokes on same topic

- Same schema instantiation but different values for MerX and SynY:
  Q. What do you call a shout with a window?
  A. A computer scream.

- A different description rule for the question specification:
  Q. What do you get when you cross a shout with a display?
  A. A computer scream.

- In STANDUP, the user can select which template the riddle should follow.



JAPE - Examples

- Some examples generated by JAPE:
  - Q. What is the difference between leaves and a car?
    A. One you brush and rake, the other you rush and brake
  - Q. How is a nice girl like a sugary bird?
    A. Each is a sweet chick.
  - Q. What is the difference between a pretty glove and a silent cat?
    A. One is a cute mitten, the other is a mute kitten.
  - Q. What do you call a strange market?
    A. A bizarre bazaar.

Shortcomings of JAPE

- No user interaction. Very few parameters could be varied.
- Exhaustive searching for words and phrases matching the schemas and templates. No way to guide the software (e.g to make jokes on a particular topic)
- No provision for approximate matches among words for similarity of sound(near-homophony)
- Proportion of intelligible jokes was very small.

STANDUP – Going Beyond JAPE

- Lexicon changes –
  - Augmented with pictures
  - Phonetic representations included (from UniSyn)
  - Restricted words to familiar words, removed obscure words (e.g vitellus, saddlery)
  - Restricted vocabulary to avoid usage of swear words etc.

STANDUP – Going Beyond JAPE

- **Removed shared roots**
  Same root word cannot appear in both question and answer (tends to spoil the joke)
  E.g Removes the following types of jokes:
  "What do you get when you cross a school principal with a rule? A principal principle."

- **Prevented excessive abstraction**
  Most words were ultimately connected via highly abstract entries such as "entity".
  Highly obscure riddles such as the one below generated.
  What do you get when you cross an aristocracy with a quality? A nobility mobility.

- Added a highly user-friendly UI. JAPE did not have one.

Computer Model of the ‘Sense of Humour’

Model of the ‘Sense of Humour’

Humour : A specific *malfunction* in the processing of information, conditioned by the necessity of a **quick deletion from consciousness of a false version**.

Model of the ‘Sense of Humour’

Consequence of the ”commutation”
of two mutually exclusive images (versions, estimates)
in the human consciousness.

Example

- Consider the following joke:
  - The horse tradesman: “If you mount this horse at 4 in the morning then at 7 in the morning you will be at Mumbai.”
  - The customer: ”But what shall I do in Mumbai at 7 in the morning?”

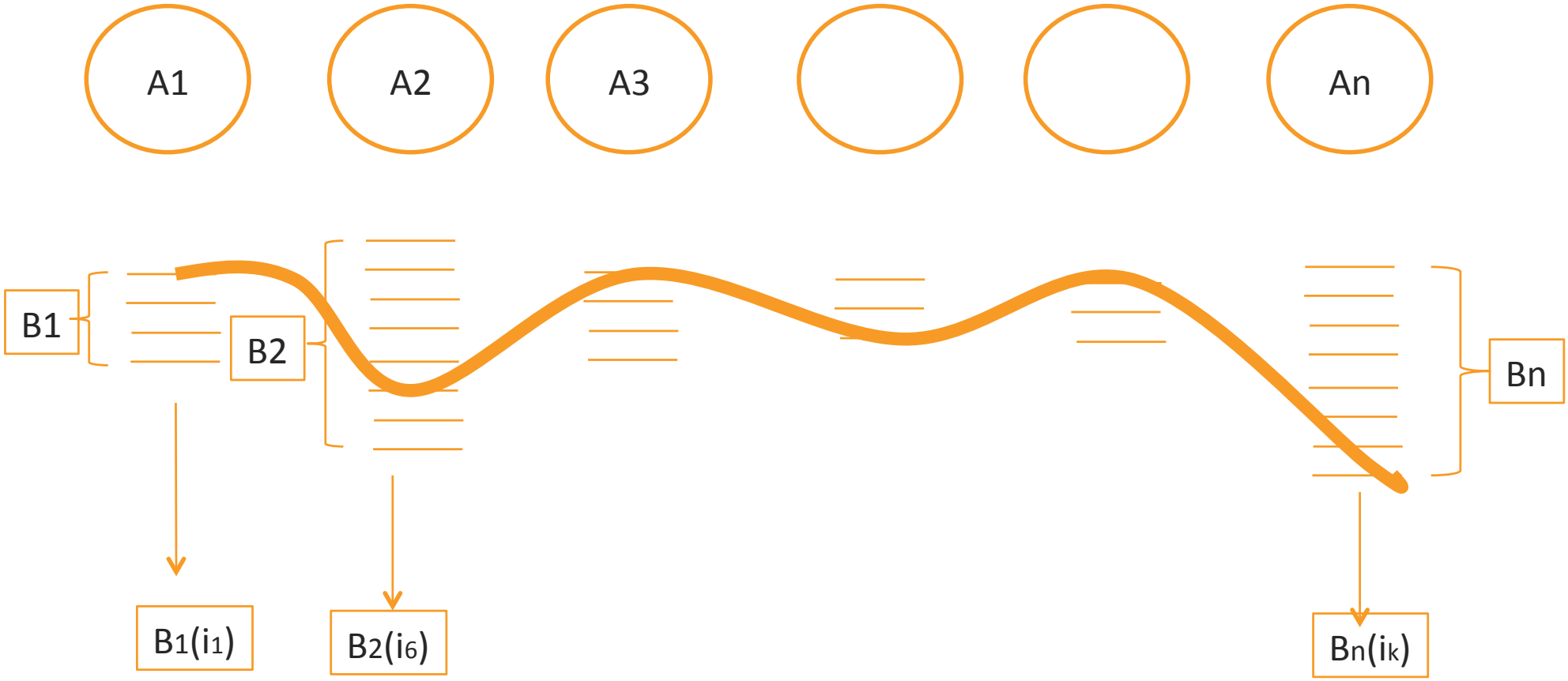
- Here the tradesman intends to tell about the horse speed but the customer thinks that he is telling how to reach Mumbai by 7 in the morning.

- This causes a commutation of two views in the brain of third person giving rise to humor in this case.

Information Processing – the Problem

- Succession of symbols A1 , A2 , A3 , . . . (”text”) is introduced from the outside world to the brain (visual or auditory)
- In the brain a set of images or meanings is put in correspondence to each word.
- The problem of information processing consists in choosing one image or meaning from the set associated with each symbol.

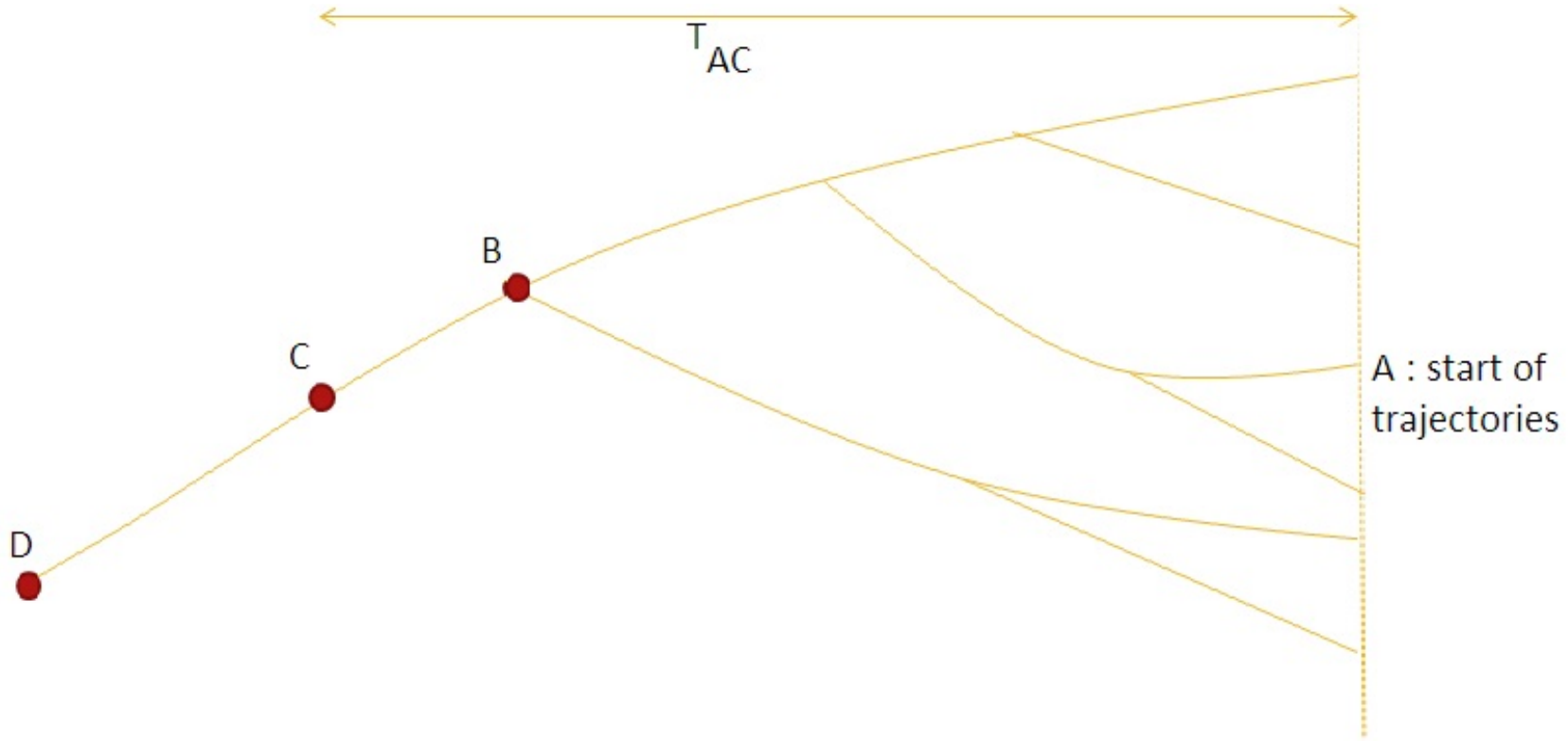
Diagrammatic Representation



The Algorithm

- The algorithm of information processing consists in the following:
  - all possible trajectories in the image space are constructed
  - a certain probability is ascribed to each trajectory on the basis of the information on the correlation of images stored in memory
  - the most probable trajectory is chosen
- Number of operations goes up exponentially with text length
- So, at any given time, only M of the most probable trajectories are conserved

Bring in the Sub-conscious

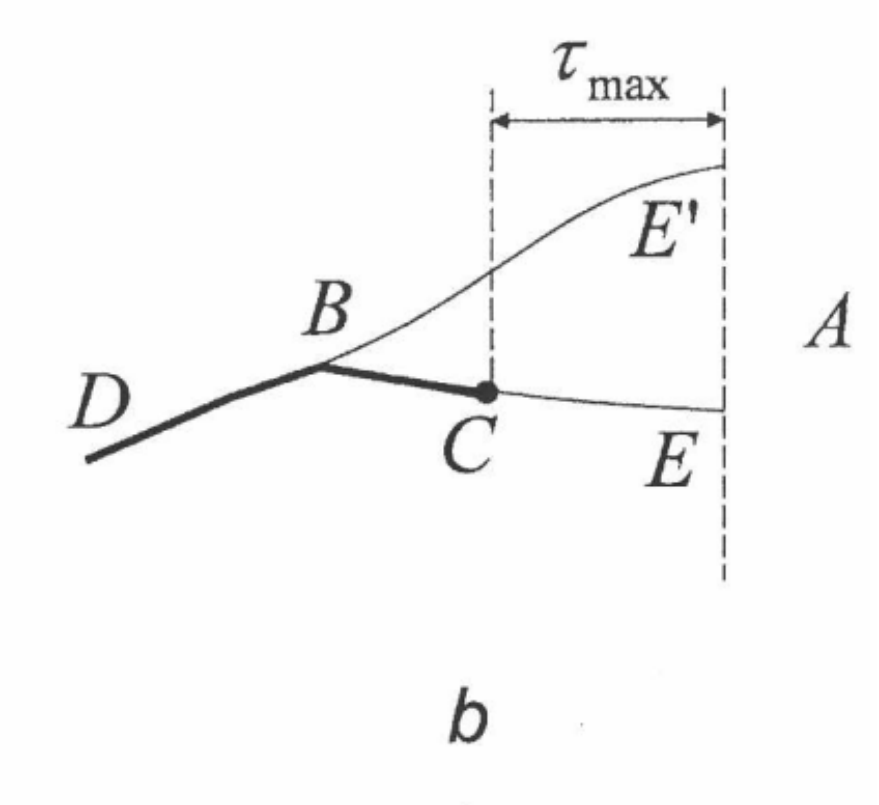


B is where all trajectories converge to a single point and we have a definite understanding at this point. C is the point where the subconscious starts sending what it understood to the conscious.

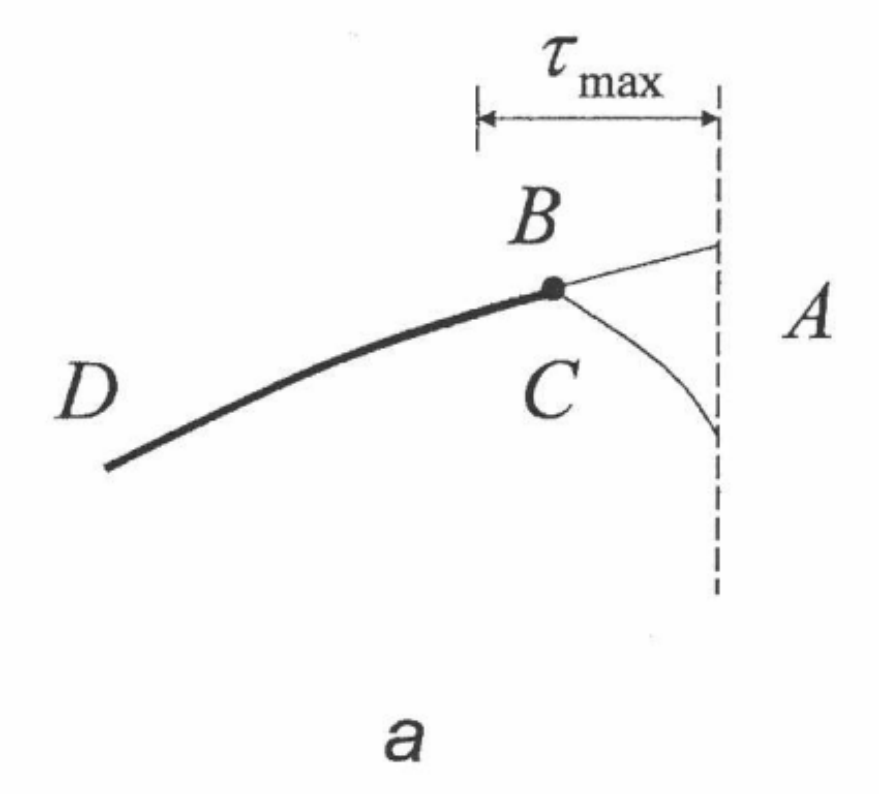
Humorous Effect

- The delay of point C with respect to front A results in the time interval T_{AC} during which the information introduced to the brain does not appear in the consciousness.
- Interval AC has upper bound τ_{max} on the time scale.
- if $T_{AB} > \tau_{max}$
  - Then $T_{AC} = \tau_{max}$ and text is interpreted incorrectly
  - Correct interpretation trajectory is sent later to consciousness causing humor sensation
- If $T_{AB} < \tau_{max}$
  - $T_{AC} = T_{AB}$ and correct interpretation is sent

Humour vs Non-Humour



b



a

Back to the Example

- When the horse tradesman says: “If you mount this horse at 4 in the morning then at 7 in the morning you will be at Mumbai.” two versions arise in subconscious, that he is telling about the speed of horse or he is telling how to reach Mumbai.

- For many people, the first version seems to be more probable.

- When the customer says “But what shall I do in Mumbai at 7 in the morning?”, the probability of second version is increased and when these two versions commute, humorous effect arises.

Correlation to Human Nature

- Different susceptibility of people to humour is connected with the differences in the delay τ_{max}.
- People with large τ_{max} seldom laugh because point C seldom outruns point B.
- Conversely, people with small τ_{max} are aware of a humorous effect even in cases that most people do not see as funny.

Applying the Suslov Model

- Computer is given a τ_{max} of a normal person
- Each word is given a set of images, trajectories are drawn, highest probable trajectory is selected and using τ_{max} humor is sensed
- Limitations
  - many options of images for a given word
  - Probability calculation of word done using determined data but is still a hard task
  - Experiments are done using ideal languages, using a real life language includes many complications

Conclusions

- Analysing humor is non-trivial. Current approaches are heavily WordNet dependent. Cannot incorporate cultural and social contexts using this approach.

- Humor changes everyday with changing societal norms and perceptions and new technological developments. E.g computer humor wouldn't have been funny 200 years ago.

- There has been definite progress. However, we still have to develop sufficient understanding of humour and of human behaviour to permit even limited forms of jokes to lubricate human computer interface
  There is still a long way to go for machines to turn into virtual humans!!

References

1. I.M.Suslov, *Computer Model of "a Sense of Humour". I. General Algorithm*. Biofizika SSSR 37, 318 (1992) [Biophysics 37, 242 (1992)]
2. Rada Mihalcea, Carlo Strapparava: *Making Computers Laugh: Investigations in Automatic Humor Recognition*. HLT/EMNLP 2005
3. Ritchie, G., Manurung, R., Pain, H., Waller, A., Black, R., & O'Mara, D. (2007). *A practical application of computational humour*. In Proceedings of the 4th International Joint Conference on Computational Creativity (pp. 91-98).
4. Pushpak Bhattacharyya, Natural Language Processing: A Perspective from Computation in Presence of Ambiguity, Resource Constraint and Multilinguality,CSI Journal of Computing, Vol. 1, No. 2, 2012.
5. http://en.wikipedia.org/wiki/Computational_humor
6. http://en.wikipedia.org/wiki/Theories_of_humour
7. http://en.wikipedia.org/wiki/Pun

Questions?

Thank you