Natural Language Processing and Applications

Unit 4

Coreference Resolution & Machine Translation

Coreference resolution: coreference phenomena - mention detection - mention-pair architecture ; RNNs for sequence labelling and classification - stacked and bi-direction RNNs - machine translation (MT): lexical divergence and typology - encoder-decoder with RNNs - MT evaluation

* Coreference Resolution

→ Coreference Resolution (CR) is the task of finding all linguistic expressions (called mentions) in a given text that refer to the same real-world entity.

→ After finding and grouping these mentions, they can be resolved by replacing pronouns with noun phrases.

eg: I voted for Trump because he was most aligned with my values", John said.

* Importance of Co-reference Resolution

→ A fundamental component of natural language understanding systems.

→ It enables machines to comprehend text by linking referring

expressions to their appropriate referents.

→ This is crucial for NLP applications such as information extraction, question answering and text summarization.

* Discourse Model

→ A discourse model is a mental representation constructed by natural language understanding systems and humans to interpret text comprehensively.

Evoking Entities : When a referent is first mentioned in the discourse, a representation for it is evoked into the model.

Accessing Entities : Subsequent mentions of the same referent access the representation already present in the model

* Anaphora

→ Refers to an entity that has already been introduced in the discourse

(i) The first mention is the antecedent

(ii) Subsequent mentions are anaphors

(iii) Entities w/ only a single mention are singletons

* Coreference Chain → A set of coreferring expressions is often called a coreference chain.

eg. {University of Illinois at Chicago, UIC, The school, it}

* Tasks in Coreference Resolution

→ Coreference resolution comprises of two key tasks:

(i) Identify referring expressions (mentions of entities)

(ii) Cluster them into coreference chains

→ Entity linking maps coreference chains to real-world entities.

* Coreference Phenomena: Linguistic Background

Referring expressions can occur in several forms:

(i) Indefinite noun phrases

(ii) Definite noun phrases

(iii) Pronouns

(iv) Proper Noans (names)

A. Indefinite Noun Phrases

→ usually marked with the determiner 'a' or 'an'

→ generally introduce new entities to the discourse.

eg. I saw this beautiful cauliflower

B. Definite Noun Phrases

→ usually marked with 'the'

→ generally refer to entities that have already been introduced in the discourse

→ Can also refer to entities that haven't been introduced to the discourse, but are identifiable to the receiver due to:

(i) world knowledge

(ii) implications from the discourse structure.

eg (i) Have you checked out the Andy Warhol exhibit?

(ii) Make sure to order the tiramisu!

c. **Pronouns**

→ Generally refer to entities that have already been introduced to the discourse and are easily identifiable

p. **Proper Nouns**

→ can be used to introduce new entities to the discourse, or to refer to those that already exist.

* Information Status

→ Referring expressions can also be categorized by their information status - the way they introduce new information or access old information.

The 3 main groups are:

(i) New noun phrases

(ii) Old noun phrases

(iii) Inferables

A. **New Noun Phrases**

① Brand new NPs : Introduce entities that are both new to the discourse and to the listener
eg. an Uber

② Unused NPs : Introduce entities that are new to the discourse but not to the listener
eg. Chicago

B. **Old Noun Phrases**

→ Introduce entities that already exist in the discourse model - and are thus not new to the discourse nor to the listener.

eg. she

C. **Inferables**

→ These introduce entities that are neither Listener-old nor discourse-old, but the listener can infer their existence by reasoning based on other entities that are in the discourse.

eg.

(i) I went to a superb restaurant yesterday. The chef had just opened it.

(ii) Mix flour, butter and water. Knead the dough until shiny.

* Linguistic Properties of the Coreference Relation

(i) Number Agreement

(ii) Person Agreement

(iii) Gender / noun class agreement

(iv) Binding theory constraints

(v) Recency

(vi) Grammatical role

(vii) Verb semantics

(viii) Selectional restrictions

A. **Number Agreement**

→ In general, antecedents and their anaphors should agree in number.

- Singular with singular
- Plural with plural

→ A few exceptions include:

(i) Some semantically plural entities (eg. companies) can be referred to using either singular or plural pronouns

(ii) "They" - can be used as a singular pronoun.

eg: IBM announced a new machine translation product yesterday. They have been working on it for 20 years

B. **Person Agreement**

→ In general, antecedents and their anaphors should agree in person

(i) First person with first person

I, my, me

(ii) Third person with third person

they, their, them

→ An exception is text containing quotations

C. **Gender / Noun Class Agreement**

→ In general, antecedents and their anaphors should agree in grammatical gender

He with his
She with hers
They with theirs

→ Knowing which gender to associate with a name in a text can be complex, and may require world knowledge about the individual.

eg.

(i) Maryam has a theorem. She is exciting

(she = Maryam, not the theorem)

(ii) Maryam has a theorem. It is exciting

(it = theorem, not Maryam)

D. **Binding Theory Constraints**

→ Antecedents and their anaphors should adhere to the syntactic constraints placed upon them.

• Reflexive pronouns (eg. herself) corefer with the subject of the most immediate clause that contains them whereas non-reflexives cannot co-refer with this subject.

eg. Janet bought herself a bottle of fish sauce [herself = Janet]

Janet bought her a bottle of fish sauce [her ! = Janet]

E. **Recency**

→ Entities introduced in recent utterances tend to be more salient than those introduced from utterances further back.

F. **Grammatical Role**

→ Entities mentioned in the subject position are more salient than those in the object position.

eg. Natalie went to the Eiffel Tower with Shahla. She took a selfie

She → Natalie

↳ **Verb Semantics**

→ Some verbs semantically emphasize one of their arguments, biasing the interpretation of subsequent pronouns

eg.

John telephoned Bill. He lost the laptop (He = John)

John criticized Bill. He lost the laptop (He = Bill)

* Coreference Tasks

→ Given a text T, find all the entities and the co-reference links between them.

Some subtasks include:

① Direct mentions

- pronominal anaphoras
  - filter out non-referential pronouns
- definite noun phrases
- indefinite noun phrases
- names

② Link those mentions into clusters

→ Annotation of Mentions and links

→ Depends on the task specifications and the dataset.

→ Some coreference datasets do not include singletons as mentions.

Identifying singletons makes the task easier. Singletons are often difficult to distinguish from non-referential noun phrases, and constitute a majority of mentions

→ Some coreference datasets provide human-labeled mentions - the task is to simply cluster those mentions into groups.

* Coreference Datasets

1. OntoNotes
2. IsNotes
3. ARRAU

* Mention Detection

→ The process of finding spans of text that constitute a referring expression (mention)

→ Many systems run parsers and NER taggers on the text and extract every span that is either an NP, a possesive pronoun, or a named entity.

→ Filtering for mention detection can be done as follows:

(1) Rule-based methods

1. Take all noun phrases, possessive pronouns, and named entities
2. Remove numeric quantities, mentions embedded in larger mentions and stop words
3. Remove non-referential 'it' based on regular expression patterns.

(2) Classifiers

→ Classifiers for mention filtering often make use of a variety of features characterizing the words, their relationship and their position in the surrounding text.

→ Some classifiers are:

(i) Referentiality classifier
(ii) anaphoricity classifier
(iii) discourse-new classifier

* Hard Filtering for Mention Detection

→ Hard filtering based on rules or classifiers isn't necessarily the best option

Filter too many → recall suffers

Filter too few → precision suffers

→ A better alternative would be to perform mention detection, anaphoricity filtering and entity clustering jointly in an end-to-end model

* Mention-Pair Architecture

Given: a pair of mentions - a candidate anaphor and a candidate antecedent

Decide: whether or not they co-refer

Working: compute coreference probabilities for every plausible pair of mentions - there will a high probability for actual

coreferring pairs, and low probability for other pairs

Learning Probabilities

1. Select training samples :

→ choose one +ve instance (m_i, m_j) where m_j is the closest antecedent to m_i

→ a negative instance (m_i, m_k) for each m_k between m_j and m_i.

2. Extract Features

→ hand built features and/or

→ implicitly learned representations

3. Train a classification model

Making Predictions

→ Apply the trained classifier to each test instance in a clustering step

Closest First Clustering -

(i) For a mention i, the classifier is run backwards through prior i-1 mentions

(ii) The first antecedent with a probability >0.5 is selected and linked to i.

Best First Clustering

(i) Classifier is run on all possible i-1 antecedents

(ii) Mention w/ the highest probability is selected as the antecedent for i.

* Advantages and Disadvantages of Mention-Pair Architecture

Advantage

→ simplest coreference resolution architecture

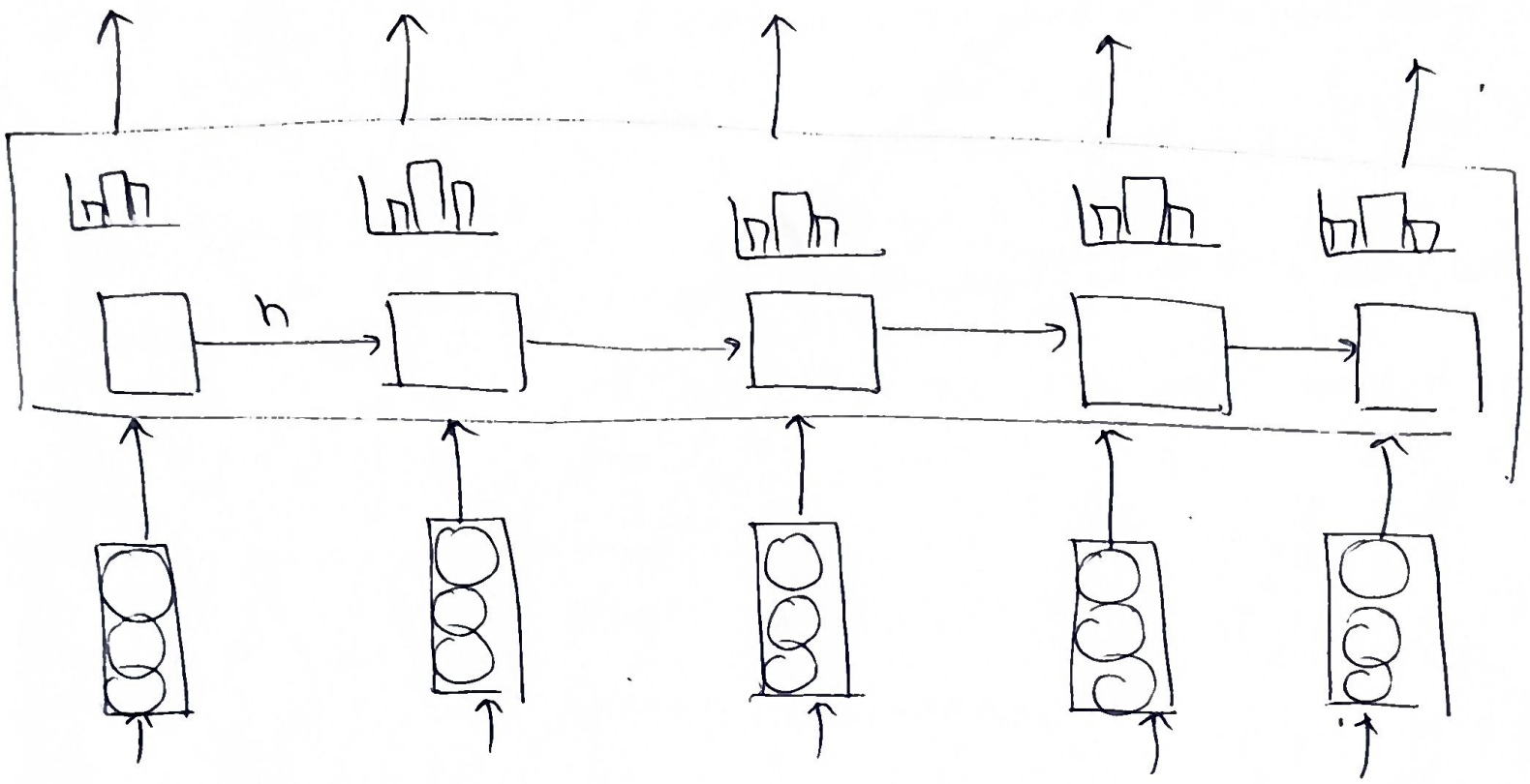
Disadvantage

→ doesn't directly compare candidate antecedents w/ one another

→ considers only mentions, not overall entities

* RNN for NLP Tasks

① RNN for sequence labelling (POS Tagging)

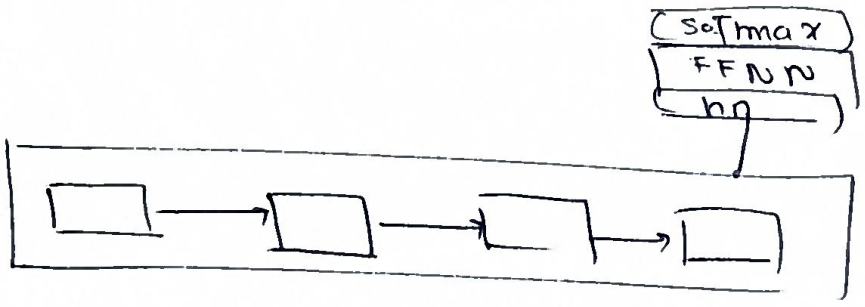


② RNNs for sequence classification

→ assign the entire sequence to a class

→ h_n for the final layer acts as a compressed representation of the entire sequence. Pass to a FFNN

→ Choose a class via softmax



→ There is no loss calculated in the intermediate stages

→ Errors are back propagated all the way back in the RNN.

* Stacked RNNs

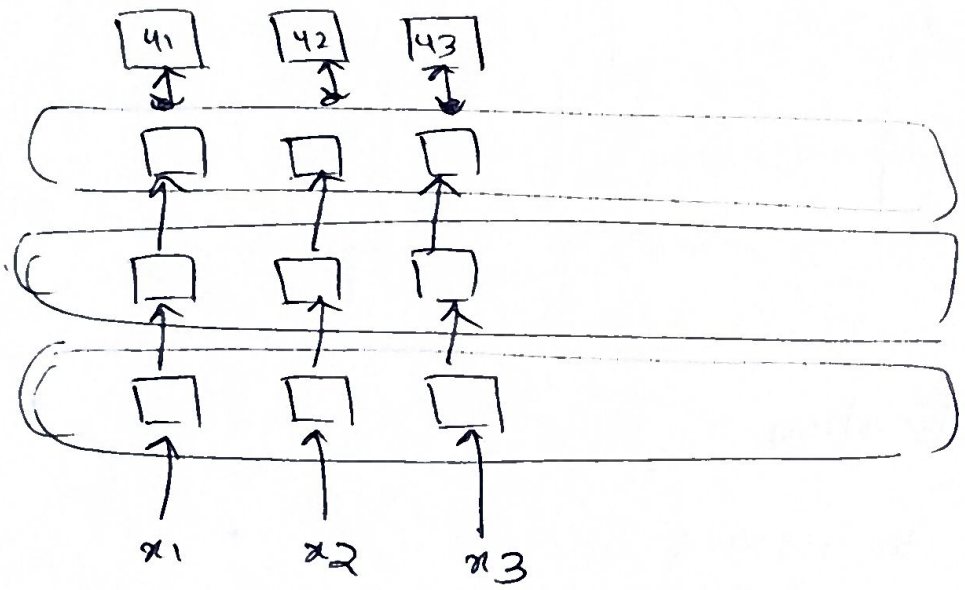
→ entire sequence of outputs from one RNN as the input sequence to the next

→ outperforms single-layer networks

→ having more layers allows the network to learn representations at different layers of abstraction

(i) Early layers → more fundamental properties

(ii) later layers → more meaningful



Optimal no. of RNNs to stack

(i) depends on application & training set

(ii) more RNNs in stack ⇒ increased training cost

* Bidirectional RNNs

→ Simple RNNs consider the info in a sequence upto the current timestep

$h_t^{f} = RNN_{forward}(x_t)$, ie to the left

→ Context to the right can also be useful.

In <u>bidirectional RNNs</u>

(i) an RNN is trained on the input sequence in reverse

$$h_t^b = RNN_{backward}(x_t^n)$$

(ii) Combine the forward and backward networks

(iii) There are thus 2 independent RNNs

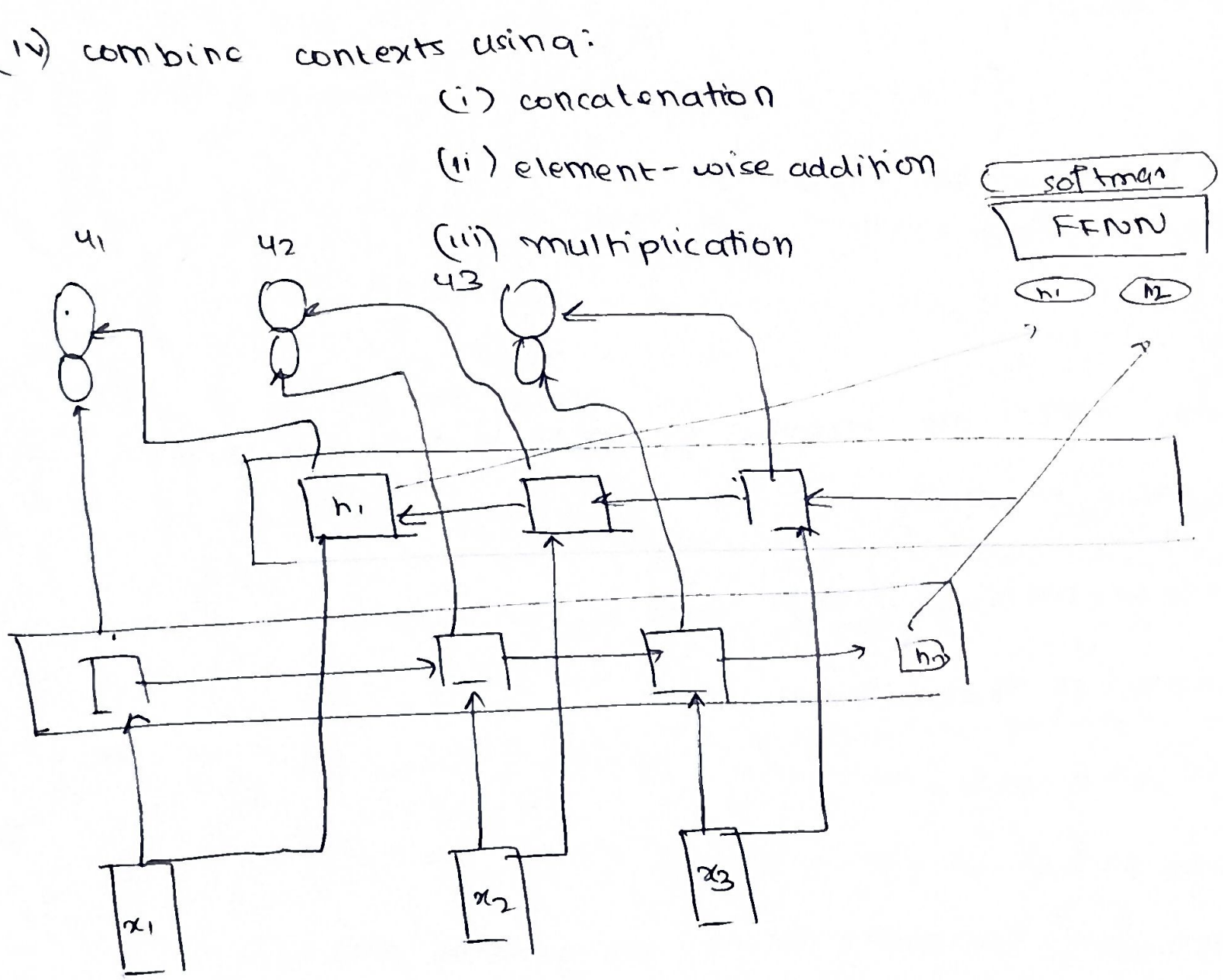
$$h_t = h_t^f \oplus h_t^b$$

(iv) combine contexts using:

(i) concatenation

(ii) element-wise addition

(iii) multiplication



* Machine Translation

→ The use of computers to automate some or all of the process of translating from one language to another

Problems w/ MT

(i) ~~Machine~~ Word Order

(ii) Word sense

(iii) Pronoun Resolution

(iv) Idioms

(v) Ambiguity

Word Order SVO US SOV

Word Sence

Idioms → idioms are composed of words that do not directly contribute to their meaning

→ Direct replacement of words leads to non-sensical translations

* Characteristics of Indian Languages

(i) SOV vs SVO order

(ii) word free order

(iii) morphological variations based on number & gender

(iv) adjects w.r.t gender

* MT Approaches

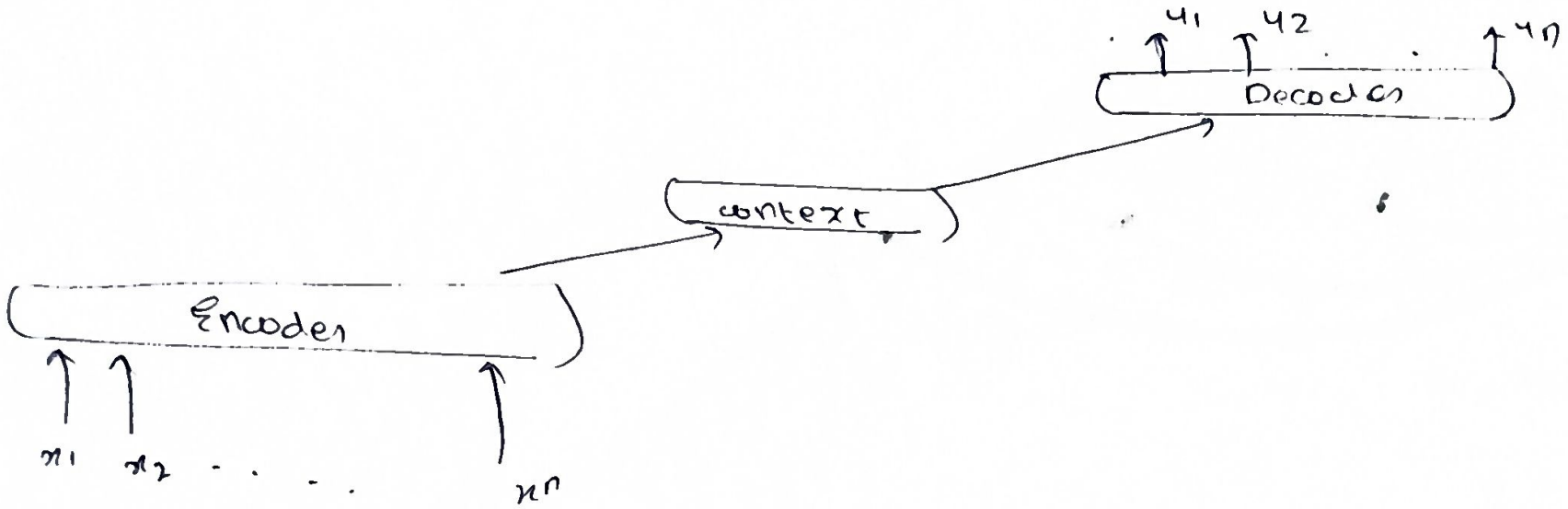
Rule-based → Transfer
→ Interlingua

Corpus-based → example
→ statistical

Knowledge-based

* Machine Translation - Encoder - Decoder Model

* **Encoder - Decoder Model** - models capable of generating contextually appropriate, arbitrary length, sequence outputs



It has 3 components:

(i) Encoder → accepts input x_1^n, and generates a corresponding contextual representations h_1^n.

→ can use LSTMs, CNNs, Transformers

(ii) context vector → c, is a function of h_1^n → conveys the essence of input to the decoder

(iii) decoder → accepts c as the input and generates an arbitrary length of hidden states h_1^m.

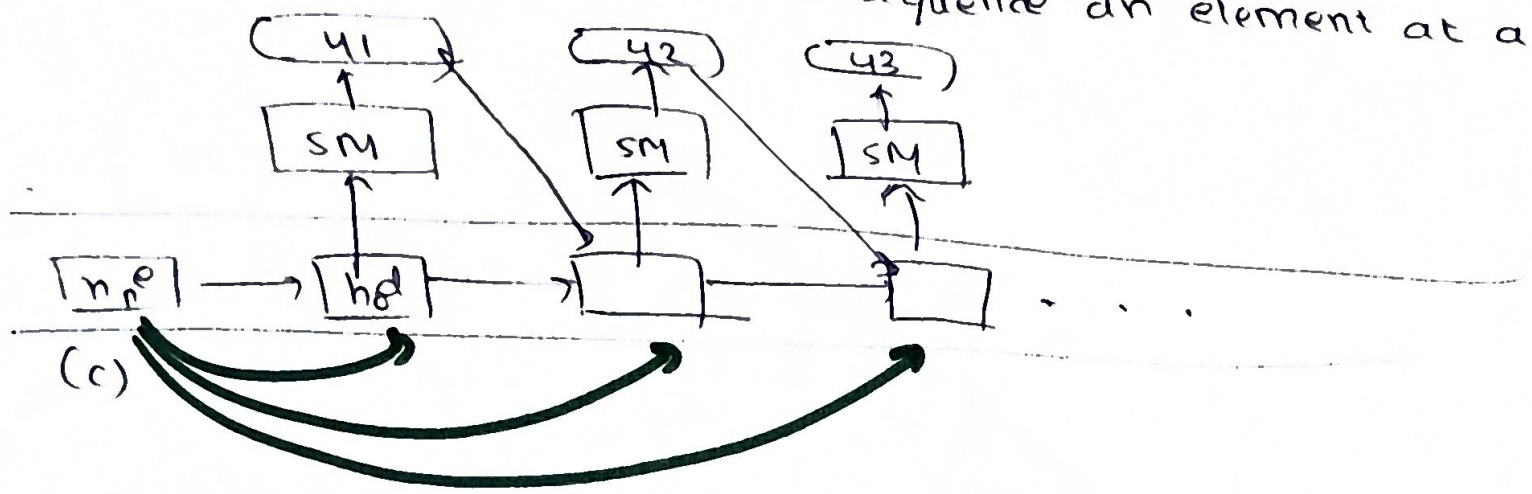
Outputs are y_1^m

Any kind of seq. architecture can be used.

* Architectural Choices for the Encoder Decoder Model

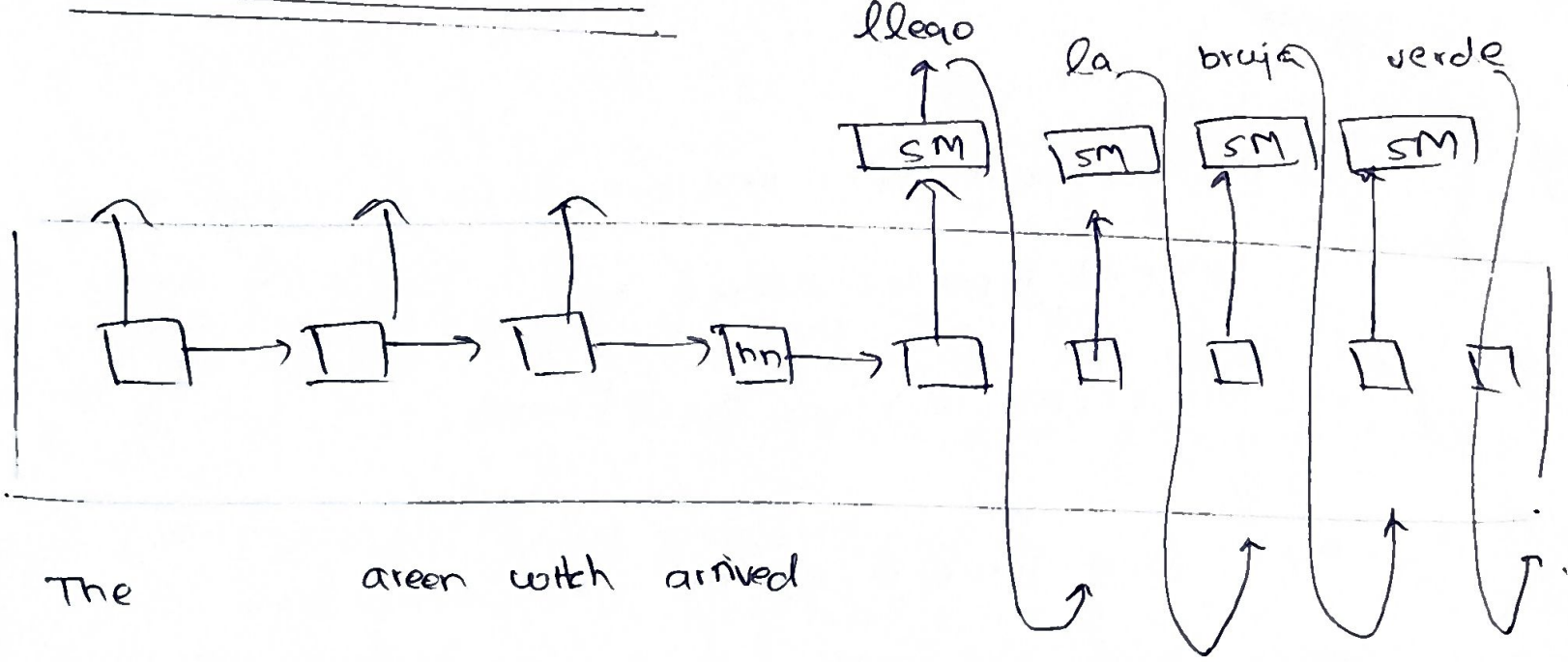
Encoder : Stacked RNNs

Decoder : Produce an output sequence an element at a time



enhancements in green

* Training the MT Model



The green with arrived

* Evaluation of the MT Model

1. Human Ratings
2. BLEU Score

Human Ratings check for fluency → naturalness, → style, → clarity
fidelity → adequacy, → informativeness

: For Fluency → rating as

→ time taken to read

Fidelity → how info was preserved

→ gold standard

→ mod

BLEU Scores → the closer the predicted sentence is to the human-generated target sentence, the better it is.

Precision = no. of corrat words / total no. of correct words

(see He He He example)

Clipped precision

→ Compare each word from the predicted sentence with all of the target sentences. If any of them match, it is considered to be correct

limit the count of each correct word to the max no. of times it occurs in the target sentence.