Statistical Machine Translation (SMT)

In SMT we aim to translate the source language sentence into a target language sentence. We attempt to solve this problem as a conditional probability of T given S. $P(T|S)$.

Now the MT System has to figure out that T that maximizes this conditional probality.

$$\hat{T} = \underset{T}{\operatorname{argmax}} (T|S)$$

Using bayes rule $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$

$$\hat{T} = \underset{T}{\operatorname{argmax}} \frac{P(S|T)P(T)}{P(S)}$$

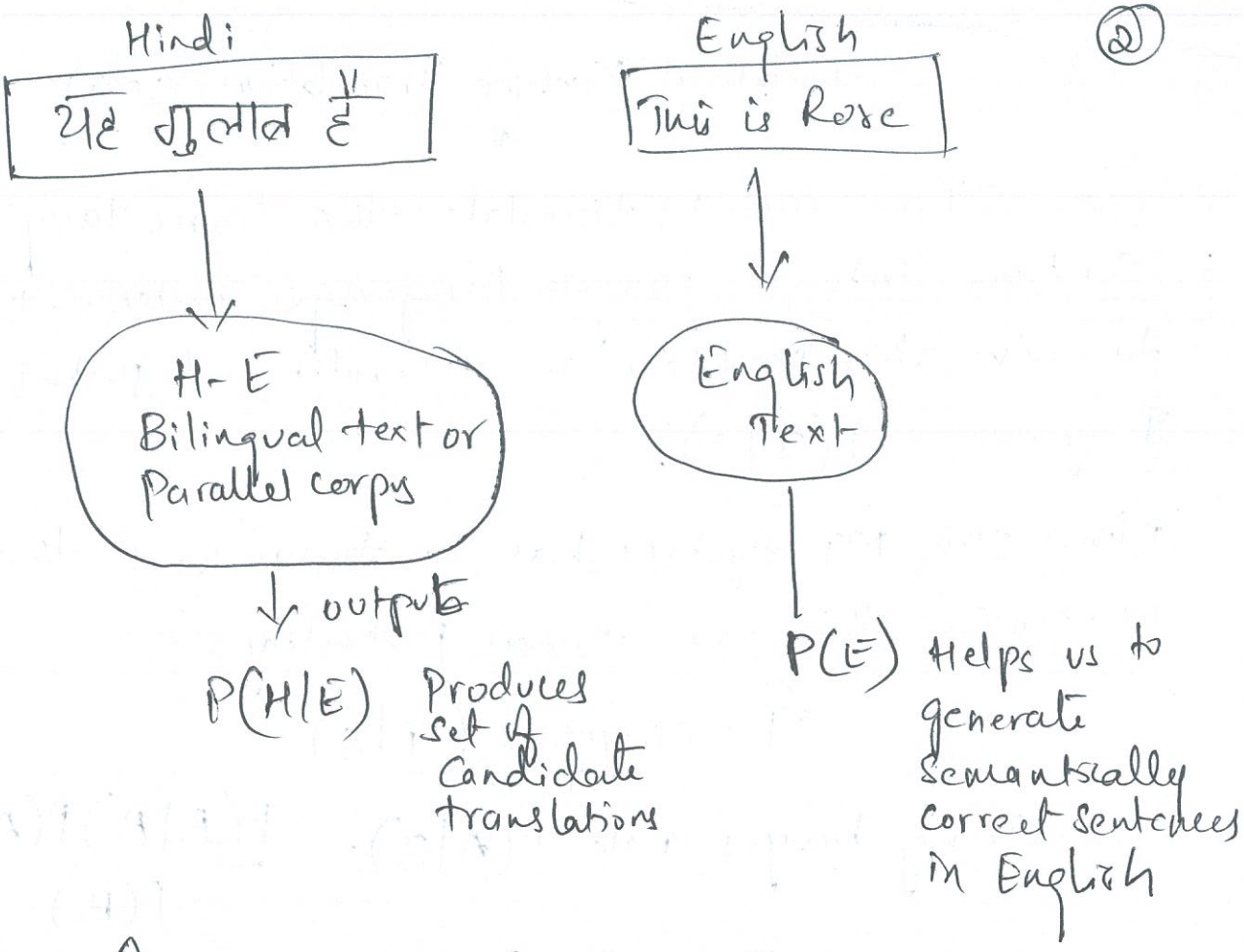
We decompose this problem into 2 subproblems and Since $P(S)$ in the denominator is not a function T and is a constant which we multiply to all T's we igore it.

$$\hat{T} = \underset{T}{\operatorname{argmax}} P(S|T)P(T)$$

$P(S|T)$ is called a translation model & is used to find the probability that S comes from T.

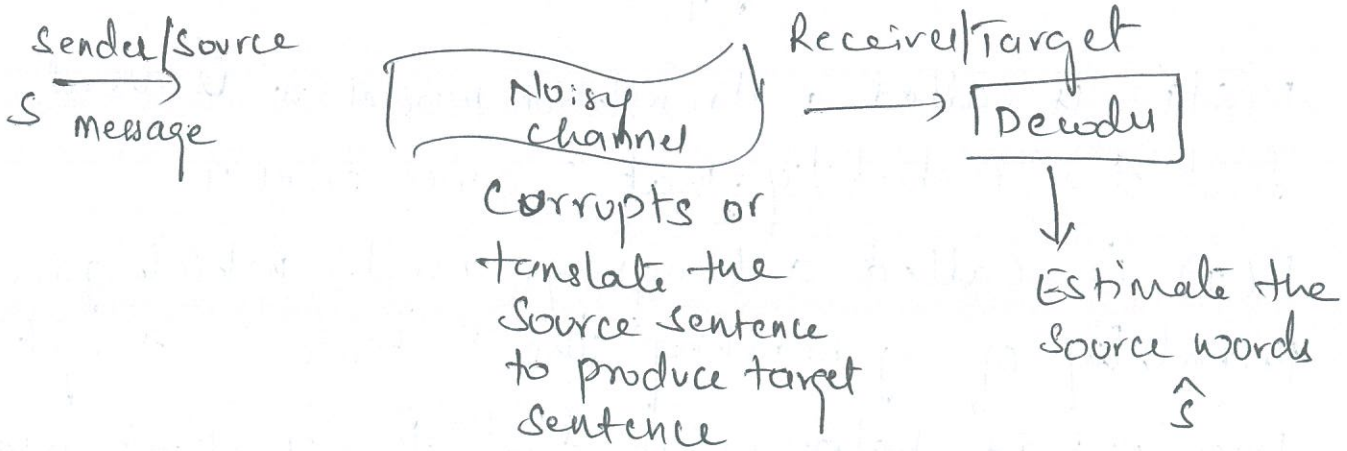
$P(T)$ is called a language model which gives us a probability of generating the sentence in target langua

Decoder: It helps us in find the $\hat{T}$ that maximizes the product of both the above subproblems.



$$\hat{E} = \underset{E}{\operatorname{argmax}}\ P(H|E)\,P(E)$$

But we have an interpretation problem in this model. Our main goal was to translate $P(T|S)$ and according to bayes rule we are estimating $P(S|T)$. The noisy channel model comes handy to say that ~~instead of~~ $P(T|S)$ is similar to $P(S|T)$



At the receiver end we need to guess the original message which was in the source language.

Now our decoding algorithm will iterate over a set of possible source sentences which could be the original message (argmax). The sentence that we picked up in source language will have a probability of being in the target language P(T). Hence whenever we multiply TM & LM we get the probability of this Target to be the original sentence of this source sentence S.

Consider that we are translating from English to Hindi originally. Now due to noisy channel model our TM will be P(E|H).

o/p sentences	P(E)	P(H|E)	P(H|E)P(E)
This rose is	low	high	low
is this rose	low	high	low
is rose this	low	high	low
This Pink is	low	high	low
Pink is good	low	low	low
This is rose	high	high	high

The argmax is a product of TM & LM and then rank all candidate target sentences based on this product and pick the one that has highest probability.

Language Model (LM): Predict the probability of generating the sentence in a given language. In our SMT we Estimate $P(T)$.

A sentence generation can be modelled as a sequence of words using chain rule.

$$P(S) = P(w_1 \cdots w_n)$$

$$= P(w_1) P(w_2|w_1) P(w_3|w_2 w_1) \cdots P(w_n|w_{n-1} w_{n-2}$$

↑ This is to lengthy

To get rid of this length subsequence we apply Markov assumption

$$P(w_i|w_{i-1} w_{i-2} \cdots w_1) = P(w_i|w_{i-1} \cdots w_{i-n-1})$$

In unigram model we assume each word is independent of each other. Hence if we are to generate a sentence of 4 words it is modelled as

$$P(w_1 w_2 w_3 w_4) = P(w_1) \times P(w_2) \times P(w_3) \times P(w_4)$$

The drawback of this model is that the word order does not matter and hence the sentences may not be semantically correct.

Bigram model

$$P(w_1 w_2 w_3 w_4) = P(w_1 | \langle \text{begin of sentence} \rangle) P(w_2|w_1) P(w_3|w_2) P(w_4|w_3$$

The probality of Each word depends only on the previous word.

Trigram model

$$P(w_1 w_2 w_3 w_4) = P(w_1 | \langle \text{begin of sentence} \rangle) P(w_2|w_1) P(w_3|w_2 w_1)$$
$$P(w_4 | w_3 w_2)$$

The longer the conditional prob window the better Syntatically correct sentences will be generated. But too longer subsequences can be generated with nearly zero probability.

How to compute these parameters $P(w_3|w_2 w_1)$?

$$P(w_3|w_2 w_1) = \frac{\text{count}(w_1 w_2 w_3)}{\text{count}(w_1 w_2)}$$

All the parameters can be computed in this form for the language model from the target language.

Now let's see how to model Translation Model

How to assign values for $P(S|T)$?

$$P(S|T) = \frac{Count(S, T)}{Count(T)}$$

These estimates needs to be done at sentence level, but this is not possible since the number of such exact sentences may be very low. Hence we model this using a hidden variable a (alignment) that represent alignment between the individual words in a sentence as marginilization over all alignments

Now let's see how to model Translation?

How do we estimate values of $P(S|T)$?

$$P(S|T) = \frac{count(S,T)}{count(T)}$$

What we are looking for is word to word mapping to Estimate these counts. If we have word aligned text we could Easily Estimate $P(S|T)$. Unfortunately we have the sentences aligned in the parallel corpus. Using these sentences we model a hidden variable a (alignment) that represent how individual words in a sentence can be mapped.

- $P(S|T)$ involves two choices
  1. choice of length l_s for the source sentence (Constraint)
  2. choice of words $s_1, s_2, s_3 \ldots s_{l_s}$

From now on we will take the length l_s to be fixed and we try to model

$$P(s_1\ s_2\ s_3 \ldots s_{l_s} | t_1\ t_2\ t_3 \ldots t_{l_t}, l_s)$$

It is difficult to Estimate this directly. Hence we introduce alignments variable. An alignment variable a identifies which target word each source word originated from.

Formally an alignment variable can takes values from $\{0, 1, \ldots l_t\}$ and its length is l_s

For Example if the translation model is $P(H|E)$

(T) E = This is rose
(S) H = यह गुलाब है

In this case the length of hindi sentence is 3

$\therefore a = \langle 1\ 3\ 2 \rangle$

(T) E = Ram hit the boy with stick
(S) H = राम ने छड़ी से लड़के को मारा

In this example length of source sentence (H) is 7 $\therefore a = \langle 1\ 1\ 6\ 5\ 4\ 3\ 2 \rangle$

Note that Each source word is aligned to Exactly one target word. This alignment is many to one i.e more than one source word can be aligned to a single target word.

Another possible alignment for first example could be

$a = \langle 1, 1, 1 \rangle$ This is rose
यह गुलाब है

This alignment is bad

Given the length of source and target sentences there would be $(l_t+1)^{l_s}$ possible alignments.

In IBM Model-1 the assumption is that all alignments of T are equally likely given the length of source.

$$\forall a,\ P(a|T, l_s) = \frac{1}{(l_t+1)^{l_s}}$$

It means that

$$P\left(\begin{array}{c}\text{This is rose}\\ \text{यह गुलाब है}\end{array}\right) = P\left(\begin{array}{c}\text{This is rose}\\ \text{यह गुलाब है}\end{array}\right)$$

Given an alignment a and the target sentence T, what is the probability of a source sentence S?

$$P(S|a, T)$$

In IBM Model-1

$$P(S|a, T) = \prod_{j=1}^{l_s} P(S_j|T_{a_j})$$

$P(S_j|T_{a_j})$ denotes the probability of j^{th} source word given that it was generated from a_j^{th} target word.

E = Ram hit the boy with stick

$a = \langle 1\ 1\ 6\ 5\ 4\ 3\ 2 \rangle$

$$P(H|a,E) = P(\text{राम}|\text{Ram})\, P(\text{ने}|\text{Ram})\, P(\text{छड़ी}|\text{stick})\, P(\text{से}|\text{with})\, P(\text{लड़के}|\text{boy})\, P(\text{को}|\text{the})\, P(\text{मारा}|\text{hit})$$

IBM-Model 1

To generate a source sentence S from target T

1. Pick a length of S
2. Pick an alignment with uniform probability

$$\frac{1}{(l_t+1)^{l_s}}$$

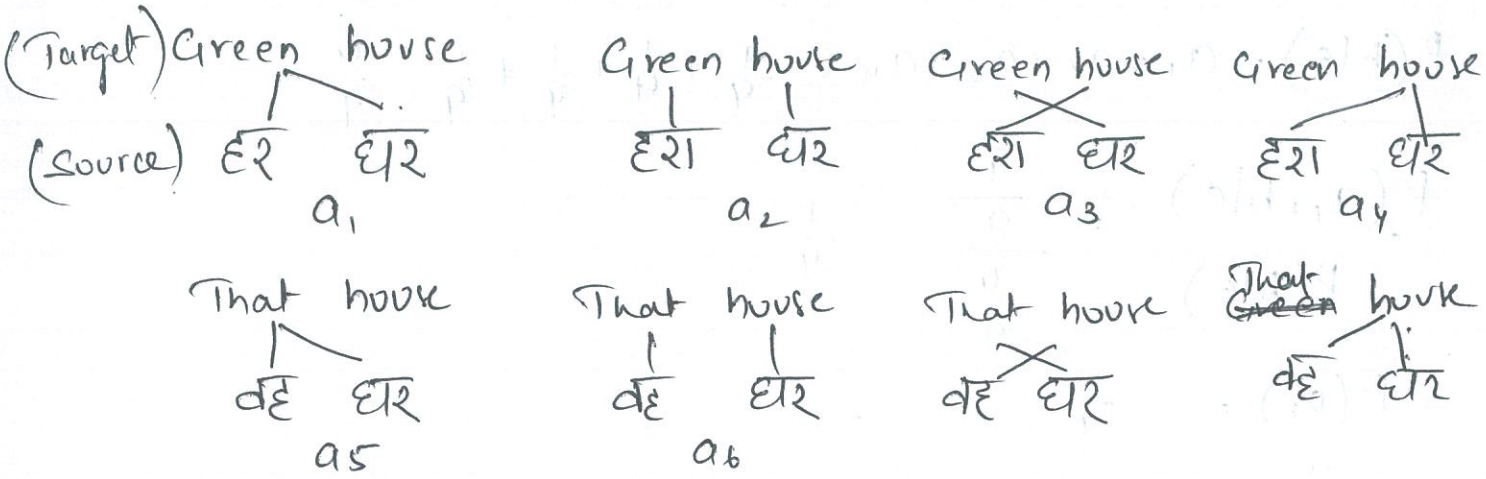
3. Sample source word with probability

$$P(S|a,T) = \prod_{j=1}^{l_s} P(S_j|T_{a_j})$$

$$\therefore P(S,a|T) = P(a|T)\, P(S|a,T)$$

$$= \frac{1}{(l_t+1)^{l_s}} \prod_{j=1}^{l_s} P(S_j|T_{a_j})$$

In order to Estimate the translation model parameters we use the Expectation maximization algorithm. Here is a worked out example with 2 sample sentences.



Initialize all parameters $t(h|e)$ uniformly

	हरा	घर	वह
Green	1/3	1/3	1/3
house	1/3	1/3	1/3
That	1/3	1/3	1/3

Compute $P(a, h|e)$ for Each alignment $a_1 - a_8$

$P(a_1, h|e) = P(\overline{\text{हरा}}|\text{Green})\, P(\overline{\text{घर}}|\text{Green}) = \frac{1}{3} \times \frac{1}{3} = \frac{1}{9}$

$P(a_2, h|e) = P(\overline{\text{हरा}}|\text{Green})\, P(\overline{\text{घर}}|\text{house}) = \frac{1}{3} \times \frac{1}{3} = \frac{1}{9}$

⋮

$P(a_8, h|e) = P(\overline{\text{वह}}|\text{house})\, P(\overline{\text{घर}}|\text{house}) = \frac{1}{3} \times \frac{1}{3} = \frac{1}{9}$

Normalize the alignment values

i.e $P(a|h,e) = \frac{P(a,h|e)}{P(h|e)}$

$P(h|e) = a_1 + a_2 + a_3 + a_4 = \frac{1}{9} + \frac{1}{9} + \frac{1}{9} + \frac{1}{9} = \frac{4}{9}$

$\frac{P(a_1,h|e)}{P(h|e)} = \frac{\frac{1}{9}}{\frac{4}{9}} = \frac{1}{4}$

$P(a_2) = \frac{1}{4}$

⋮

$P(a_8) = \frac{1}{4}$

Compute fractional counts

$c(\text{इस}, \text{Green}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (using a_1, a_2)

$c(\text{इस}, \text{Green}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (a_1, a_3)

$c(\text{ये}, \text{Green}) = 0$ (These words never occured in any alignment)

$c(\text{इस}, \text{house}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (a_3, a_4)

$c(\text{इस}, \text{house}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (a_2, a_4)

$c(\text{ये}, \text{house}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (a_7, a_8)

$c(\text{इस}, \text{that}) = 0$

$c(\text{इस}, \text{that}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (a_5, a_7)

$c(\text{ये}, \text{that}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$ (a_5, a_6)

total count (Green) $= \frac{1}{2} + \frac{1}{2} + 0 = 1$

(house) $= \frac{1}{2} + \frac{1}{2} + \frac{1}{2} = \frac{3}{2}$

(that) $= 0 + \frac{1}{2} + \frac{1}{2} = 1$

Maximization step

$t(\overline{ह्रा} | \text{Green}) = \frac{\frac{1}{2}}{1} = \frac{1}{2}$

$t(\overline{घर} | \text{Green}) = \frac{\frac{1}{2}}{1} = \frac{1}{2}$

$t(\overline{वह} | \text{Green}) = 0$

$t(\overline{ह्रा} | \text{house}) = \frac{\frac{1}{2}}{\frac{3}{2}} = \frac{1}{3}$

$t(\overline{घर} | \text{house}) = \frac{\frac{1}{2}}{\frac{3}{2}} = \frac{1}{3}$

$t(\overline{वह} | \text{house}) = \frac{\frac{1}{2}}{\frac{3}{2}} = \frac{1}{3}$

$t(\overline{ह्रा} | \text{that}) = 0$

$t(\overline{घर} | \text{that}) = \frac{\frac{1}{2}}{1} = \frac{1}{2}$

$t(\overline{वह} | \text{that}) = \frac{\frac{1}{2}}{1} = \frac{1}{2}$

Keep iterating over the Expectation & Maximization steps till all the translation parameters converge.

IBM Model-2

In Model-1 all alignment are equally likely which is wrong. Since languages having SOV (Subject Object Verb) order like most Indian languages the verbs predominantly move to the end of the sentence. Once the position of the noun is fixed adjectives qualify these nouns. Hence it is realistic to assume a probability distribution over position of words as

$$q(i|j, l_t, l_s)$$

q can be interpreted as the probability that j^{th} source word is connected to i^{th} target word, given the sentence lengths of source & target sentence as l_t and l_s respectively.

$$P(s_1, s_2 s_3 \ldots s_{l_s}, a_1 a_2 \ldots a_{l_s} | t_1 t_2 \ldots t_{l_t}, l_s) =$$

$$\prod_{j=1}^{l_s} q(a_j|j, l_t, l_s)\, T(s_j|t_{a_j})$$

Target : 1 That 2 house 3 is 4 beautiful

Source j वह घर सुंदर है

$a = \langle 1\ 2\ 4\ 3 \rangle$

$$P(S|T) = t(\text{वह}|\text{that})\, q(1|1,4,4) \times$$
$$t(\text{घर}|\text{house})\, q(2|2,4,4) \times$$
$$t(\text{सुंदर}|\text{beautiful})\, q(4|3,4,4) \times$$
$$t(\text{है}|\text{is})\, q(3|4,4,4)$$

IBM Model-3

The assumption in IBM Model 1 & 2 is that Each word in the Target generates one or many source words but their distribution is uniform whether it generates one or many words. This factor is handeld as a fertility factor in Model 3. For Each target word t in target sentence $\bar{T}$, we choose a fertility ϕ. The choice of ϕ depends on t which is modeled as a probability distribution parameter $n(\phi|t)$. These parameters for Each target word is computed from the training corpus.

We are given $\bar{T} = \{t_1, t_2, t_3 \ldots t_{l_t}\}$ want to model $P(S|\bar{T})$.

Since we have l_t no of target words we have to compute l_t+1 fertilities with probability

$$P(\{\phi_0, \phi_1, \phi_2 \ldots \phi_{l_t}\} | \bar{T})$$

where ϕ_i stands for the no of source words t_i generates and $n(\phi_i|t_i)$ is the probability that t_i generates ϕ_i words.

For Example given the following Target Sentence

That house is beautiful.

we compute a probability distribution

$n(0|\text{that}) = 0.1$ $n(0|\text{house}) = 0.001$ $n(0|\text{is}) = 0.1$ ...

$n(1|\text{that}) = 0.9$ $n(1|\text{house}) = 0.9$ $n(1|\text{is}) = 0.9$

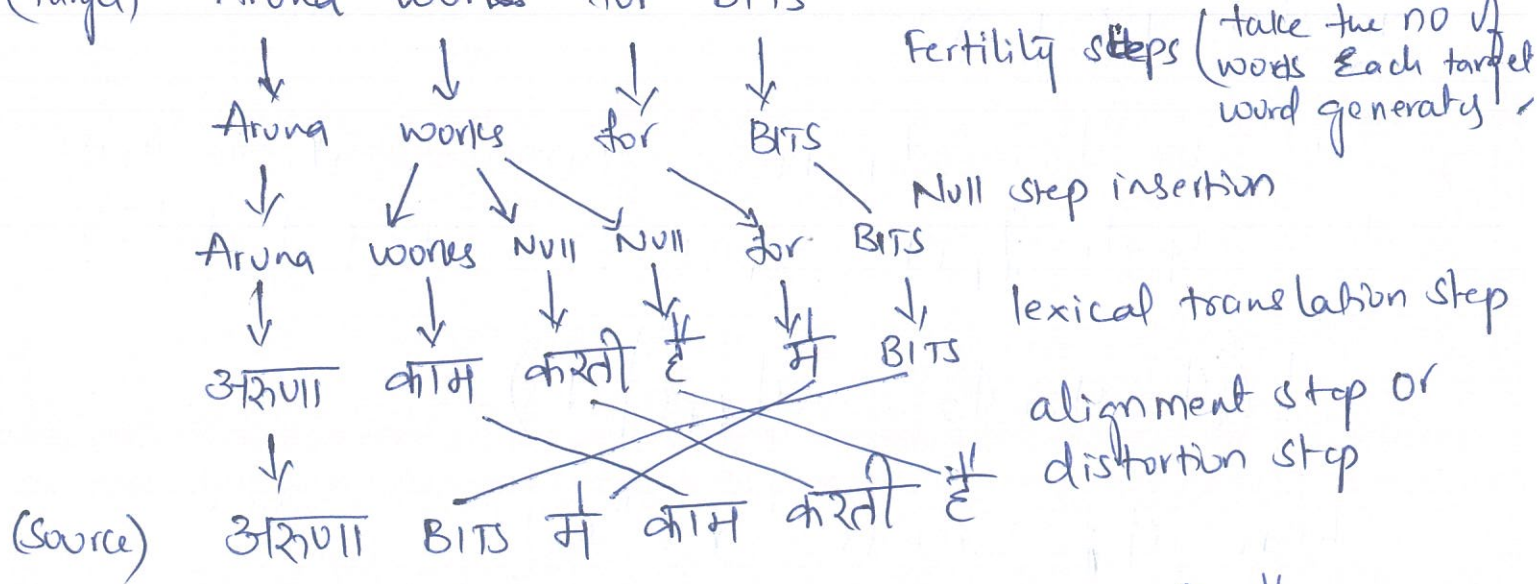
$n(2|\text{that}) = 0$ $n(2|\text{house}) = 0.09$ $n(2|\text{is}) = 0$

⋮ ⋮ ⋮

Choose $l_t + 1$ fertilities $\{\phi_0 \dots \phi_{l_t}\}$ with probability

$$P(\{\phi_0, \phi_1, \phi_2 \dots \phi_{l_t}\} | t) = \underbrace{P(\phi_0 | \phi_1, \phi_2, \dots \phi_t)}_{\text{Null word insertion}} \prod_{i=1}^{l_t} n(\phi_i | n_i)$$

Lets see an Example of the whole process of translation



If we want to add the source words करती है #, it can be generated by the special Null token.

We could model the fertility of the Null token in the same way as all other words by the conditional distribution $n(\phi|\text{Null})$. However the no of inserted words clearly depends on the sentence length, hence we model the Null word insertion as a special step.

∴ we represent a parameter P_0 which is the probability of having no Null token inserted after Every target word t. P_1 is the probability of generating one Null token inserted after Each target word t.

The number of source words generated from a Null token is represented as ϕ_0

$$\text{No of o/p words} = \sum_{i=1}^{l_t} \phi_i = l_s - \phi_0$$

The estimation of ϕ_0 is treated as a binomial distribution since Each target word t may generate a null (P_1) or not (P_0). $P(\phi_0)$ is the probability of generating ϕ_0 source words from a Null token is Estimated as

$$P(\phi_0) = \binom{l_s - \phi_0}{\phi_0} P_1^{\phi_0} P_0^{l_s - 2\phi_0} \quad \left(\begin{array}{c}\text{using binomial}\\ \text{distribution}\end{array}\right)$$

$$P(S|T, l_s) = \binom{l_s - \phi_0}{\phi_0} p_1^{\phi_0} p_0^{l_s - 2\phi_0} \prod_{i=1}^{l_t} \phi_i! \, n(\phi_i | t_i) \times$$

$$\prod_{j=1}^{l_s} q(a_j | j, l_t, l_s) \, t(s_j | t_{a_j})$$