A phrase is a continuous sequence of words. The basic idea of phrase-based translation is to segment the given source sentence into phrases, then translate each phrase and finally compose the target sentence from these phrase translations. State-of-the-art for many language pairs and used by GoogleTranslate and others.

The following are the advantages of phrase-based translation

- Translating phrases helps to reduce translation ambiguities
- Phrases of arbitrary length: sometimes the entire sentence might be covered by a phrase
- Simpler model: no more need to explicitly model the concepts of fertility, insertion and deletion of words

Mathematical Definition

Let us now define the phrase-based statistical machine translation model mathematically. First, we apply the Bayes rule to invert the translation direction and integrate a language model pLM. Hence, the best English translation e_{best} for a foreign input sentence f is defined as

$$e_{best} = p(E|F)$$
$$= \text{argmax}_e\ p(F|E)\ p(E)$$

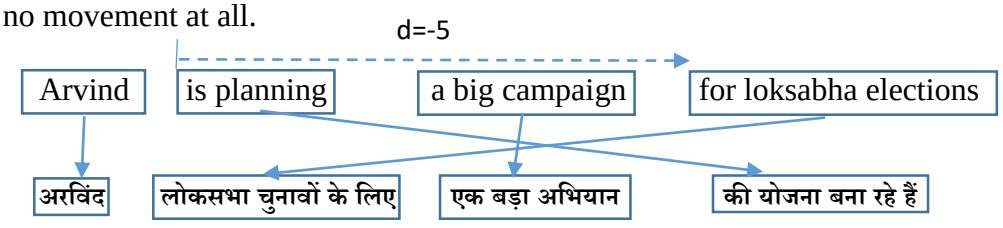
This is exactly the same reformulation that we have already seen for word-based models. For the phrase-based model, we decompose p(F|E) further into

$$p(\bar{f}_1^I|\bar{e}_1^I) = \prod_{i=1}^{I} \phi(\bar{f}_i|\bar{e}_i)\ d(\text{start}_i - \text{end}_{i-1} - 1)$$

Each English phrase e_i is translated into a foreign phrase f_i. Note that this process of segmentation is not modeled explicitly. This means that any segmentation is equally likely.

Reordering is handled by a distance-based reordering model. The distance-based reordering model consider reordering relative to the previous phrase. We define $start_i$ as the position of the first word of the English input phrase that translates to the i^{th} Foreign phrase, and end_i as the position of the last word of that English phrase. Reordering distance is computed as $start_i - end_{i-1} - 1$.

The reordering distance is the number of words skipped (either forward or backward) when taking English words out of sequence. If two phrases are translated in sequence, then $start_i = end_{i-1}+1$; i.e., the position of the first word of phrase i is the same as the position of the last word of the previous phrase plus one. In this case, a reordering cost of d(0) is applied. What is the probability of d? Instead of estimating reordering probabilities from data, we apply an exponentially decaying cost function $d(x) = \alpha^{|x|}$ with an appropriate value for the parameter $\alpha \in [0, 1]$ so that d is a proper probability distribution. This formula simply means that movements of phrases over large distances are more expensive than shorter movements or no movement at all.



Phrase in foreign(Hindi)	Translates from English word positions	Distance travelled by i^{th} English phrase using $d= start_i - end_{i-1} - 1$ to produce the corresponding Foreign Phrase
1	1	1-0-1=0
2	7-9	7-1-1=5
3	4-6	4-9-1=-6
4	2-3	2-6-1=-5

In IBM Model 2, each target (Foreign) word is aligned to exactly one English word. The matrix shows these alignments for P(F|E).

	अरविंद (a)	लोकसभा (b)	चुनावों (c)	के (d)	लिए (e)	एक (f)	बड़ा (g)	अभियान (h)	की (i)	योजना (j)	बना (k)	रहे (l)	हैं (m)
arvind (1)	■												
is (2)													■
planning (3)										■	■	■	
a (4)						■							
big (5)							■						
campaign(6)								■					
for (7)				■	■								
loksabha (8)		■											
elections (9)			■										

Fig: Each target (Foreign) word is aligned to exactly one English word. The matrix shows these alignments for P(F|E).

	अरविंद (a)	लोकसभा (b)	चुनावों (c)	के (d)	लिए (e)	एक (f)	बड़ा (g)	अभियान (h)	की (i)	योजना (j)	बना (k)	रहे (l)	हैं (m)
arvind (1)	■												
is (2)													■
planning (3)										■			
a (4)						■							
big (5)							■						
campaign(6)								■					
for (7)					■								
loksabha (8)		■											
elections (9)			■										

Fig: Each target (Foreign) word is aligned to exactly one English word. The matrix shows these alignments for P(E|F).

Three ways of combining these alignments into phrases exist in phrasebased SMT. The process is called symmetrization.

1. Intersection: A = A1 ∩ A2.
2. Union: A = A1 ∪ A2.
3. Intersection-Union combination
    a. A = (A1 ∩ A2) ∪ A1
    b. A = (A1 ∩ A2) ∪ A2

	अरविंद (a)	लोकसभा (b)	चुनावों (c)	के (d)	लिए (e)	एक (f)	बड़ा (g)	अभियान (h)	की (i)	योजना (j)	बना (k)	रहे (l)	हैं (m)
arvind (1)	■												
is (2)													■
planning (3)										■			
a (4)						■							
big (5)							■						
campaign(6)								■					
for (7)					■								
loksabha (8)		■											
elections (9)			■										

A phrase-pair consists of a sequence of source (English) words, paired with a sequence of target (Foreign) words,

• A phrase-pair (e, f) is consistent if:

– There is at least one word in e aligned to a word in f

– There are no words in f aligned to words outside e

– There are no words in e aligned to words outside f

• Extract all consistent phrase pairs from the training example

	अरविंद (a)	लोकसभा (b)	चुनावों (c)	के (d)	लिए (e)	एक (f)	बड़ा (g)	अभियान (h)	की (i)	योजना (j)	बना (k)	रहे (l)	हैं (m)
arvind (1)	■												
is (2)													■
planning (3)										■			
a (4)						■							
big (5)							■						
campaign(6)								■					
for (7)					■								
loksabha (8)		■											
elections (9)			■										

For any phrase pair (f,e) extracted from the training data, can calculate:

$$t(f|e) = \frac{\text{count}(\text{f}, \text{e})}{\text{count}(\text{e})}$$

T(एक बड़ा अभियान | a big campaign) = count(एक बड़ा अभियान , a big campaign)
count(a big campaign)

Giving Different Weights to Model Parameters

The e_{best} is found by combining P (E) and P(F|E). The advantage of the generative method is that it cleanly separates the adequacy and fluency dimensions of the translation problem. Generative approaches are founded on the maximum likelihood principle. Find parameters such that the data or observation likelihood is maximized. In the present case the data or the observation is the parallel corpora.
The parameters are:
1. The language model probabilities (n-gram probabilities)
2. Phrase translation probabilities
3. Distortion probabilities

These parameters are then used to score the candidate translations of an input sentence. There are translation situations in which we need to give different weightages to these parameters. For example, when the source and target languages are very close to each other, the reordering requirement is minimal. Then the distortion parameter should be given a low weightage.
This can be arranged if we reformulate our problem by assigning different weights for each of the above 3 parameters as follows:

By adding weights, we are guided more by practical concerns than by mathematical rigor. However, we do come up with a model structure that is well known in the machine learning community: a log-linear model.
Log-linear models have the following form:

$$p(x) = \exp \sum_{i=1}^{n} \lambda_i h_i(x)$$

• number of feature function n = 3;

• random variable x = (e, f , start, end);

• feature function h1 = log ϕ;

• feature function h2 = log d;

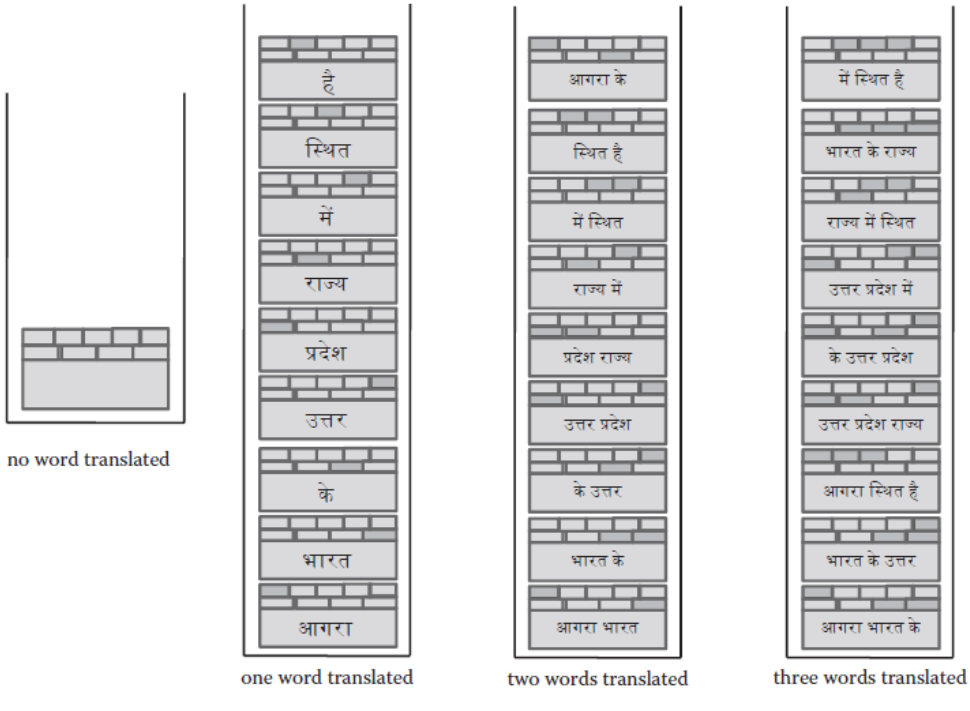
• feature function h3 for language model = log p.

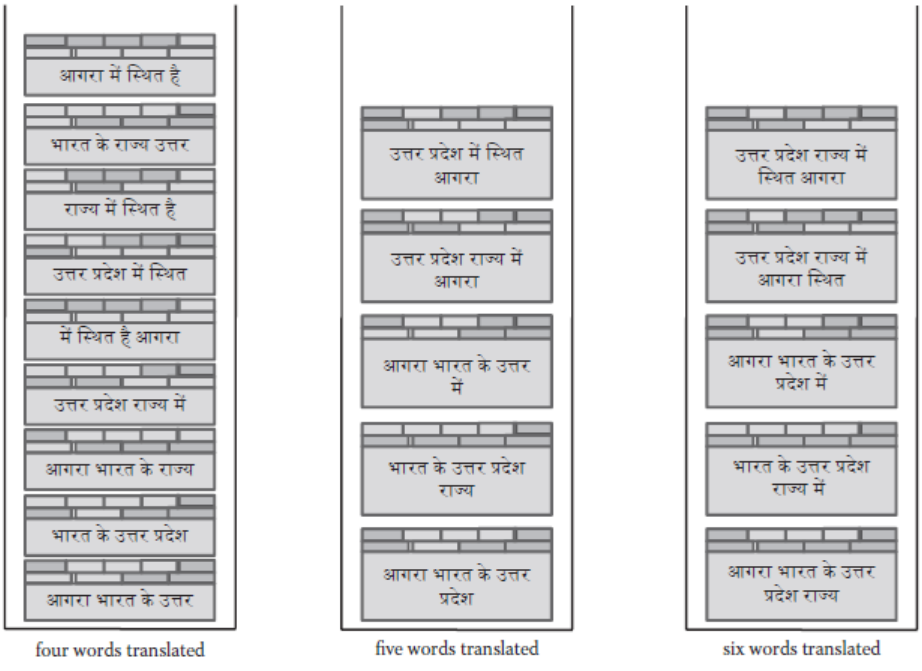
$$= \arg\max_e [\prod_{i=1}^{I} \{\Phi(\overline{f_i} \mid \overline{e_i})^{\lambda_\phi} \; d(start_i - end_{i-1} - 1)^{\lambda_d}\} P_{LM}(e)^{\lambda_{LM}}]$$

$$= \arg\max_e \left(\exp \left[\lambda_\phi \sum_{i=1}^{I} \Phi(\overline{f_i} \mid \overline{e_i}) + \lambda_d \sum_{i=1}^{I} \log d(start_i - end_{i-1} - 1) + \lambda_{LM} \log P_{LM}(e) \right] \right)$$

where λ_ϕ, λ_d, and λ_{LM} are the weightage parameters of translation model, distortion model, and language model, respectively.

Decoding





How to measure the quality of machine translation?

Given the F,E pair we seek humans to translate the given F to E.

Hindi: बिल्ली चटाई पर बैठ गई

English Reference Translation 1: the cat sat on the mat
English Reference Translation 2: there is a cat on the mat

Since we are using two humans to translate both translation are fine.

Bleu(Bilingual Evaluation Under study) score allows you to automatically compute a score for an automatically translated sentence which indicate how good the translation is. The intuition is that the machine translations must be close to human translations. Any sentence that is close to human translations will receive a high score.

The BLEU score takes a machine translated sentence and looks at the type of words it generates appear at least in one of the human generated sentences. The human generated sentences are provided as part of the test set.

Precision = number of overlapping words/ total number of words in the SMT output

For example if the SMT output is: the the the the the the the

Then the precision is 7/7= 1 which is high but the translation is extremely poor.

Modified Precision = count(max number of times it appears in the reference) / number of words in the SMT
=2/7

This is a unigram model.
In case of bigram model
English Reference Translation 1: the cat sat on the mat
English Reference Translation 2: there is a cat on the mat
SMT output: The cat the cat on the mat

	Count number of times the bigrams appear in SMT (a)	Count number of max number times the bigrams appear in References (b)	Modified Bigram precision
The cat	2	1	Sum of b/sum of a =4/6
cat the	1	0	
cat on	1	1	
on the	1	1	
the mat	1	1	

The precision is the amount of overlap between reference to the SMT

ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It is essentially of a set of metrics for evaluating automatic summarization of texts as well as machine translation. It works by comparing an automatically produced summary or translation against a set of reference summaries (typically human produced).Let us say, we have the following system and reference summaries:

Example:

Hindi: बिल्ली चटाई पर बैठ गई

English Reference Translation 1: the cat sat on the mat
English Reference Translation 2: there is a cat on the mat

For example if the SMT output is: the the the the the the the

If we consider just the individual words, the number of overlapping words between the system summary and reference summary is 1. This however, does not tell you much as a metric. To get a good quantitative value, we can actually compute the precision and recall using the overlap.
Precision and Recall in the Context of ROUGE Simply put, Recall in the context of ROUGE means how much of the reference summary is the system summary recovering or capturing? If we are just considering the individual words, it can be computed as:
Recall = number of overlapping words/ total number of words in the reference summary

Recall = 1/6

SMT output: The cat the cat on the mat: Recall = 4/6

This means that all the words in the reference summary has been captured by the system summary, which indeed is the case for this example.