NLP1 Assignment 4

Ujjwal Sharma

15 December 2016

1 PMI

(a) In the first case P(x,y) represents the probability that the first word of a bi-gram is "eat" and and the second word "pizza".

In the second case, P(x,y) refers to the probability where "happy" and "tweet" both occur at any place in a tweet.

(b)
- A **negative** PMI is the result of picking words where there is a high chance of individual occurrence but a low(non-zero) chance of occurring together. For example connectives like 'this', 'and' might have very high counts but only occur together in one bi-gram 'this and hence the entropy function is negative.
- The PMI is **zero** when two words have are absolutely independent of each other and have no correlation causing their joint probability of occurence to be equal to the product of the their individual probabilities of occurrence. For example Food and Poop will generally have no bearing on each other and will be independent.
- A **positive** PMI is for words that commonly occur together in bi-grams as they do individually like 'San Francisco'.

2 MaxEnt Models

In the MaxEnt expression,

$$P(y|x) = \frac{1}{Z} exp \sum_i w_i f_i(\overrightarrow{x}|y)$$

$$logP(y|x) = -logZ + \sum_i w_i f_i(\overrightarrow{x}|y)$$

Now

$$Z = \sum_y exp \sum_i w_i f_i(\overrightarrow{x}|y)$$

Alternatively, We can write

$$logZ = log \sum_y exp \sum_i w_i f_i(\overrightarrow{x}|y) = K$$

Hence we get,

$$logP(y|x) = -logZ + \sum_i w_i f_i(\overrightarrow{x}|y)$$

$$logP(y|x) = \sum_i w_i f_i(\overrightarrow{x}|y) - K$$

The model is log-linear in nature.

$\sum_y exp(\sum_i w_i f_i(\vec{x}, y)) = 44.9$
$K =$

- y = 1
  $\sum_i w_i f_i(\vec{x}, y) = 2 - 0.1 = 1.9$
  $P(y = 1|\vec{x}) = 0.149$

- y = 2
  $\sum_i w_i f_i(\vec{x}, y) = 1.8 + 1.1 = 2.9$
  $P(y = 2|\vec{x}) = 0.404$

- y = 3
  $\sum_i w_i f_i(\vec{x}, y) = 2 - 0.1 = 1.9$
  $P(y = 3|\vec{x}) = 0.447$

Since the third sense has the highest probability, it is the most probable.

3 First Order Logic

(a) Bears are Furry.

(b) Jan helps Joost

(c) Sergii ate the pizza

(d) Sergii ate the pizza with a fork.

(e) Every student who lifts, lifts Marie.

(f) There exists at-least some student who lifts, he lifts Marie.

4 First Order Logic

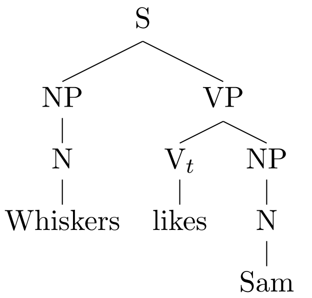
(a) $\exists_e hating(e, x) \wedge hater(e, Juan) \wedge hated(e, Pasta)$

(b) $\exists_x x.student(x) \wedge \forall_y classes(y) \rightarrow likes(x, y)$

(c) $\exists_x x.student(x) \wedge sees(e, Mary) \wedge seenBy(e, Mary)$

5 Semantic Attachments

Parse tree for sentence is



The parse from semantic attachments are:

$$VP.sem = V_t.sem(NP.sem)$$

$$VP.sem = \lambda_x.\lambda_y.\exists_e Liking(e) \wedge Liker(e, y) \wedge Likee(e, x)(Sam)$$

$$VP.sem = \lambda_x.\lambda_y.\exists_e Liking(e) \wedge Liker(e, y) \wedge Likee(e, Sam)$$

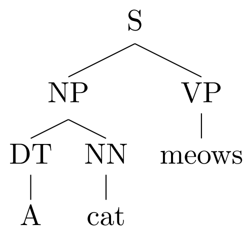
$$NP.sem = N.sem = \lambda_x.\lambda_y.\exists_e Liking(e) \wedge Liker(e, y) \wedge Likee(e, Sam)$$

$$NP.sem = N.sem = \lambda.x.\lambda_y.\exists_e Liking(e) \wedge Liker(e, y) \wedge Likee(e, Sam)(Whiskers)$$

$$NP.sem = N.sem = \lambda.x.\lambda_y.\exists_e Liking(e) \wedge Liker(e, Whiskers) \wedge Likee(e, Sam)$$

$$N.sem = \exists_e Liking(e) \wedge Liker(e, Sam) \wedge Likee(e, Whiskers)$$

(a) In this expression the term A cat is not an instance but a specific class term in itself of the type $\exists_x Cat(x)$ that is an existential quantifier on the cat.



(b) If we add the rule NP→ DT NN , DT→A to the existing grammar rule set,

S → NP.sem(VP.sem) → NP VP
NP → DT.sem(NN.sem) → DT NN

$V_p \rightarrow V_i.sem \rightarrow V_i$
$V_i \rightarrow \lambda_x \exists_m eowing(e, x) \wedge meower(x)$
Combining into $NP \rightarrow DTNN$

$= \lambda_P \lambda_Q \exists_x P(x) \Rightarrow Q(x) \lambda_x cat(x) = \lambda_Q \exists_x cat(x) \Rightarrow Q(x)$

$$\exists_x cat(x) \Rightarrow \exists_e Meowing(e) \wedge Meower(e, x)$$

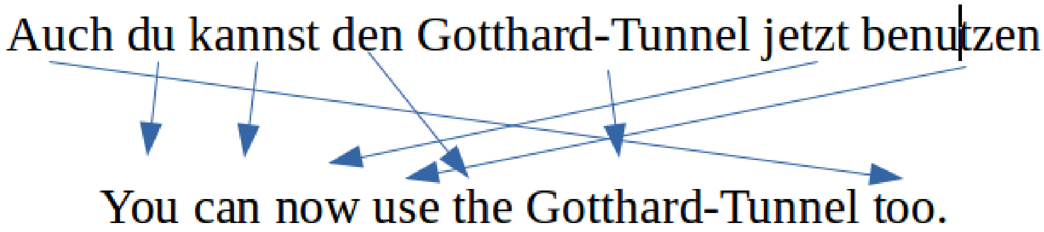
6 IBM 1 Model

The IBM 1 model makes three general simplifying assumptions:

i. The model assumes that all connections are equally likely. This implies that in general the ordering of words in the parallel English and French corpus has absolutely no impact on their translation to the other language. This assumption breaks up into two distinct independence assumptions:

   - It is assumed that the alignment links are independent given the French sentence length m and the English sentence.
   - French words are conditionally independent given the English sentence and their alignment links.

   This assumption is particularly weak since most languages where words follow each other would have typically different structures after translations but IBM 1 makes no assumptions as to the probabilities of any of these structures and considers all of them equally likely. This is primarily due to an effort to keep IBM 1 simple.
   For example,



ii. There is a uniform distribution over all possible alignments making them equi-probable and hence in concept providing no discriminative feature to suggest the suitability of one model over the other. This assumption is weak since it generally makes no assumptions to any word-ordering and was consequently corrected in IBM model 2 which still places significant focus on the placement of words while translating.

iii. The problem of fertility also arises wherein a certain translation may create multiple words or no words at all after translation. This problem is addressed by the fertility of models. This is common in languages

with different structures like German and Persian. To address this issue language model commonly places a NULL character at the start and hence does not model it probabilistically.
For example, the german sentence below creates two words for zum.