Text Entailment and Machine Translation Evaluation

Shubham Gautam

June, 2014

1 Introduction

The recognition of textual entailment is one of the recent challenges of the Natural Language Processing (NLP) domain and one of the most demanding. Indeed, as specified in (Johan Bos, 2005), recognizing entailment bears similarities to Turing's famous test to assess whether machines can think, as access to different sources of knowledge and the ability to draw inferences seem to be among the primary ingredients for an intelligent system. Moreover, many NLP tasks have strong links to entailment: in Summarization (SUM), a summary should be entailed by the text; Paraphrases (PP) can be seen as mutual entailment between a text T and a hypothesis H; in Information Extraction (IE), the extracted information should also be entailed by the text; in Question Answering (QA) the answer obtained for one question after the Information Retrieval (IR) process must be entailed by the supporting snippet of text.

Machine Translation is a very important task and to evaluate the output of the translation system, a metric is required which can be reliable in terms of correlation with human judgement. There are various existing metrics such as: BLEU, METEOR, TER *etc.* but are found inadequate in quite a few language settings like, for example, in case of free word order languages. Every metric has its own characteristic.

2 Definition

A text T is said to entail a textual hypothesis H if the truth of H can be inferred from T. Textual entailment (TE) in natural language processing is a directional relation between text fragments. The relation holds whenever

the truth of one text fragment follows from another text. In the TE framework, the entailing and entailed texts are termed text (t) and hypothesis (h), respectively. Textual entailment is not the same as pure logical entailment- it has a more relaxed definition: "t entails h"(t —> h) if, typically, a human reading t would infer that h is most likely true. The relation is directional because even if "t entails h", the reverse "h entails t"is much less certain. (Dagan et al., 2005)

Following is the example which may be helpful to understand the concept of "Text Entailment":

T : Green cards are becoming more difficult to obtain.
H : Green card is now difficult to receive.

Entailment : YES

This example is taken from RTE 1 development set. How one can say that the hypothesis is entailed by the given text? There are various approaches by which one can determine the result of entailment. In this report, the complete framework of "Text Entailment"such as the application of this beautiful concept in many important techniques will be presented.

Classical Definition: A text t entails hypothesis h if h is true in every circumstance of possible world in which t is true.

This definition is very strict since it requires truthfulness of h in all the instances where t is true. Due to uncertainties in the real world applications, this definition is not very helpful. Hence applied definition of Textual Entailment is presented:

Applied Definition: A text t entails hypothesis h if human reading t will infer that h is **most likely** true.

Again, this definition is abstract for systems trying to implement Textual Entailment. Thus mathematically precise and computable definition using probabilities is provided:

Mathematical Definition: Hypothesis h is entailed by text t if

$$P(h \; is \; true \mid t) > P(h \; is \; true) \tag{1}$$

$P(h \; is \; true \mid t)$ is the Entailment Confidence and can be considered as a measure of surety of entailment.

3 Entailment Triggers

There are various triggers that are crucial in detecting entailment. Some of these triggers along with the examples are mentioned here:

Synonymy: It is a very common entailment trigger. Text entails Hypothesis if one of the word in the Text is replaced by its synonym in the Hypothesis.

T: *India won the world cup in 2011.*

H: *India got the world cup in 2011.*

Hypernymy/Hyponymy: Entailment is also affected by the generalization or specialization of the concepts.

T: *Ram ate breads.*

H: *Ram ate food. (Hypernymy or Generalization)*

T: *He is interested in a game.*

H: *He is interested in cricket. (Hyponymy or Specialization)*

Holonymy/Meronymy: Part-for-Whole or Whole-for-part concepts can also lead to entailment. For example,

T: *Barack Obama visited Mumbai.*

H: *Barack Obama visited India. (Holonymy or Part-for-Whole)*

T: *British left India.*

H: *British left south India. (Meronymy or Whole-for-part)*

Co-reference: Co-reference is one of the main trigger for text entailment. Usually long sentences contain co-references which need to be resolved in order to determine entailment.

T: *Michael Dell announced a new strategy for the company. He is the founder of Dell.*

H: *Michael Dell is the founder of Dell.*

Quantifiers: Quantifiers play very important role in Text Entailment. Its very crucial to handle quantifiers such as *no, few, some, many, almost, all, every, etc.* while recognizing entailment.

T: *Every employee must file income tax return.*

H: *An employee must file income tax return.*

T: *Few parrots flew over the fence.*

H: *All parrots flew over the fence. (Incorrect Entailment)*

Modality: It contains the notion of *possibility* or *necessity* which may lead to incorrect entailment.

T: *This movie may break all the records.*

H: *This movie breaks all the records. (Incorrect Entailment)*

Polarity: Polarity also plays an important role when the fact is asserted or its negation occurs. Polarity can be affirmative or negative. Judging entailment based on the polarity is a difficult task since in many cases truthfulness or falsehood of the hypothesis cannot be judged.

T: *Justine denies that he ate chocolate pie.*

H: *Justine ate chocolate pie. (Entailment unknown)*

T: *Steve hates to go to work everyday early in the morning.*

H: *Steve goes to work everyday.*

Factivity: The context in which a verb phrase is used may carry semantic pre-supposition. There are some factive verbs such as *knows, learn, remembers, regrets* and *realized*, which pre-suppose the factual truth about their objects.

T: *John regrets after cheating him in the game.*

H: *John cheated him in the game.*

Comparisons: Comparison between two concepts implies similarity between them at some level. This similarity may lead to entailment in some cases.

T: *Jack is a cook but not better than Sid.*

H: *Sid is a cook.*
Here, *Jack* is a cook but not better than *Sid* in turn implies that both of them are cook.

Sequence and Order: It is vital to keep track of sequence and ordering of the events occuring between the events.

T: *Ram has gone for shopping after reading.*

H: *Ram was reading before going for shopping.*

Passivization: Active and passive voice express the same meaning. Presence of passive voice of the *text* in the *hypothesis* may lead to entailment.

T: *TCS announced a grand party to its employees.*

H: *A grand party was announced by TCS for its employees.*

Conventions: Conventions play a crucial role in many entailment cases. These include mathematical, scientific, geographical or day-to-day life conventions. The meaning of sentences which rely on such conventions cannot be determined without knowing the way the convention is applied in the sentence.

T: *Mahatma Gandhi (October 2, 1869 – January 30, 1948) was the prominent leader of Indian nationalism in British-ruled India.*

H: *Mahatma Gandhi was born on October 2.*

Numeric calculations: Certain level of numeric calculations along with the conventions can affect the entailment.

T: *Aryabhata(476-550 CE) wrote Aryabhatiya in 499 CE.*

H: *Aryabhata wrote Aryabhatiya when he was 23 years old.*

4 Approaches to Text Entailment

There are different approaches for determining entailment between two sentences (in entailment terminology: text and hypothesis). In the following sections, text entailment approaches are discussed.

4.1 Bag of Words

In this model (MacCartney, 2009), a text (such as a sentence or a document) is represented as an unordered collection of words. In this approach, every word from the hypothesis is compared with every word of the given text. If there is a match of words between T and H upto some predefined threshold then result is given as "Entailment", Otherwise "No Entailment".

It ignores the syntax and even the word order of the input sentences, and makes no attempt at semantic interpretation. The model depends only on some measure of lexical similarity between individual words. The precise

similarity function used is not essential to the model, the choice of similarity function can be viewed as a model parameter.

Approach

Let, $P(h|t)$ denotes the probability that text t supports an inference to (entails) hypothesis h. Suppose that the probability that a given word in h is independent of whether any other word in h is supported in the given text t. So, we can have the probability for entailment as follows:

$$P(h|t) = \prod_j P(h_j|t)$$

Consequently, the probability that a given word in h is supported by t can be identified with the max over the probability of its support by the individual words of t. Thus,

$$P(h_j|t) = \max_i P(h_j|t_i)$$

Thus, we can express the overall probability as follows:

$$P(h|t) = \prod_j \max_i P(h_j|t_i)$$

The expression $P(h_j|t_i)$ can be interpreted as a sort of lexical entailment score between words t_i and h_j. This is the final formula that gives the entailment between two sentences.

4.2 Natural Logic

This is the kind of logic which is purely based on "Natural Languages". In other words, we can say that it does not contain operators such as: $\neg, \wedge, \vee, \forall, \exists$ like other kinds of logic. It only contains words and phrases.

There are many kinds of inferences which are **not captured by *natural logic*** such as:

- **Temporal Reasoning** (dealing with time)
- **Causal Reasoning** (Causality is the relationship between an event *the cause* and a second event *the effect*, where the second event is understood as a consequence of the first)
- **Paraphrase** (A paraphrase is a restatement of the meaning of a text or passage using other words)
- **Relation Extraction** (to derive the relation between two entities given in the sentence)

4.2.1 Monotonicity

In the context of *entailment* "Monotonicity"refers to the fact that while adding a formula (word) to the text (T), if the hypothesis (H) is still entailed by T, then we can say that monotonicity is followed.

In other words, it can also be defined as the property of many logical systems that states that the hypothesis of any derived fact may be freely extended with additional assumptions.

Given a function f of functional type $\langle \alpha, \beta \rangle$:

Upward Monotonicity

f is upward-monotone ($\uparrow$) iff for all $x, y \in \alpha$,
$x \sqsubseteq y$ entails $f(x) \sqsubseteq f(y)$.
Example:

$$\begin{array}{rcl} \text{tango} & \sqsubseteq & \text{dance} \\ \text{and, Paris} & \sqsubseteq & \text{France} \\ \text{So, tango in Paris} & \sqsubseteq & \text{dance in France} \end{array}$$

Downward Monotonicity

f is downward-monotone ($\downarrow$) iff for all $x, y \in \alpha$,
$x \sqsubseteq y$ entails $f(y) \sqsubseteq f(x)$.
Example:

$$\begin{array}{rcl} \text{tango} & \sqsubseteq & \text{dance} \\ \text{but, didn't dance} & \sqsubseteq & \text{didn't tango} \end{array}$$

Non Monotonicity

f is non-monotone (#) iff it is neither upward nor downward-monotone.

4.3 Lexical Entailment

There are various branches of *text entailment*, one of them is **Lexical Entailment**. The task of *lexical entailment* is to determine the entailment between a pair of sentences on the basis of only *lexical concepts*.

Example	Text	Hypothesis
1	Ram is a hindi speaker	Ram speaks hindi
2	Ram was born in India	
3	Shyam was born in Mumbai	Shyam's birthplace is Mumbai
4	Shyam has grown up in Mumbai	

Table 1: Sentence pairs

4.3.1 Probabilistic Lexical Model

From table 1, we can see that for first pair of examples, one can easily say that if the given text is : *Ram is a hindi speaker* then the given hypothesis will definitely hold. But, if the text is : *Ram was born in India* then the hypothesis may or may not be true. It will be true with a certain probability depending upon the context of which a speaker is talking about (because India is a country where various languages are spoken, so it should be clearly specified in the sentence that of which part of India, the speaker is talking about). Likewise, the second pair of examples state the condition.

So, there arises the need for determining the probability of a hypothesis given the text (it is similar to the *conditional probability*).

Let, the prior probability of a hypothesis is $P(h)$. So, the concept of *text entailment* is relevant only when $P(h) < 1$, because for $P(h) = 1$ there is no need for considering the *text* (*in such case, hypothesis will be TRUE always*). (Glickman and Dagan, 2005)

Let, T = space of possible texts
$t \in T$ be a specific text
H = space of all possible hypotheses
$h \in H$ be a specific hypothesis

A text t probabilistically entails a hypothesis h (denoted as $t \rightarrow h$) if t increases the likelihood of h being true, *i.e.*,

$$P(Tr_h = 1|t) > P(Tr_h = 1)$$

where, Tr_h is the random variable for a hypothesis

4.4 Lexical Entailment Model

A hypothesis is assumed to be true if and only if all its lexical components are true as well (Glickman and Dagan, 2005). This is the main theme behind *lexical entailment*.

Let, u be a term in hypothesis h and it is assumed that the truth probability of each term in h is independent of that of the other term, *i.e.*, it follows the *I.I.D.*[1] *property.*

Thus, we obtain:

$$P(Tr_h = 1|t) = \prod_{u \in h} P(Tr_u = 1|t)$$
$$P(Tr_h = 1) = \prod_{u \in h} P(Tr_u = 1)$$

Let, $t = \{v_1, v_2.....v_n\}$ and assume that the term u from h is aligned to the most probable word from t.

$$P(Tr_u = 1|t) = max_{v \in t} P(Tr_u = 1|Tr_v)$$

where, T_v is the event that a generated text contains the word v.

Thus, the *entailment probability* based on the *lexical entailment probability* will be as follows:

$$P(Tr_h = 1|t) = \prod_{u \in h} max_{v \in t} P(Tr_u = 1|Tr_v)$$

But, this approach does not fulfill the expectation of a user because it is dependent only on the lexical state of a sentence and not dealing with syntactic and semantic based approach.

4.5 Machine Learning Based Approaches

Machine Learning (ML) is an important branch of Artificial Intelligence (AI). Machine learning is the science of getting computers to act without being explicitly programmed. Machine Learning can be applied in every branch of AI. In Natural Language Processing (NLP), it plays a vital role. Now, the question of interest is: How Machine Learning can be applied to the recoginition of *text entailment*? In TE framework, it can be applied using its classifiers to train the data and then test the accuracy on the test data.

Support Vector Machine[2] is mainly used as the classifier, it may be due to its property that it ignores the data points that are far from the inter-boundary of the regions and only take into account the data points that are nearer or on the seperator-boundary of the regions. There are many other classifiers which are used here such as: linear classifiers, logistic regression *etc.*

Machine Learning methods are used when there is a large data-set of text-hypothesis pair and also corresponding Gold-Standard results are also

[1] Independent and Identically Distributed

[2] more commonly known as SVM in ML community

Feature sets	*features*	IE	IR	QA	SUM
similarity measures on **words**	10	X	X	X	X
similarity measures on **stems**	10	X	X	X	X
+ similarity measures on **POS tags**	+10	X	X		
+ similarity measures on **chunk tags**	+10	X			X
+ **average** of sim. measures on **words** of best **partial** match	+1				X
+ **average** of sim. measures on **stems** of best **partial** match	+1			X	X
+ **average** of sim. measures on **POS tags** of best **partial** match	+1			X	X
+ **average** of sim. measures on **chunk tags** of best **partial** match	+1		X		X
+ similarity measures on **words** of best **partial** match	+10				
+ similarity measures on **stems** of best **partial** match	+10				X
+ similarity measures on **POS tags** of best **partial** match	+10	X			
+ similarity measures on **chunk tags** of best **partial** match	+10				
+ **negation**	+2	X			
+ **length ratio**	+1	X			
+ similarity measures on **nouns**	+10	X			
+ similarity measures on **noun stems**	+10				
+ similarity measures on **verbs**	+10				X
+ similarity measures on **verb stems**	+10				
+ **short/long** T	+1	X		X	
Total	128	64	31	23	54

Figure 1: Feature sets chosen in each subtask (Prodromos Malakasiotis, 2007)

provided then a classifier can be trained on the data-set to capture the features. This data-set can also be tuned by providing some of the training data as test set so that the classifier to be used can capture the main features of the training data to be effective.

As per (Prodromos Malakasiotis, 2007), if there are many tasks to be performed then for each subtask, one classifier (SVM) can be used. Suppose there are 4 subtasks in which training data belongs to: *QA, IR, IE, SUM*[3], so while training the corresponding classifier, only those kind of pairs should be used for getting accurate results. Because every subtask has its unique feature on which it mainly depends on. By classifying the training data and applying classifiers to those pairs, the classifier checks only those features reducing the time of training, tuning and testing of data.

From figure 1, we can see that different features were selected for each subtask.

[3]Summarization

4.6 Graph Matching

Graph Matching is that approach for *text entailment* in which every sentence is represented by a directed graph. In that graph, each word or phrase is represented by a *node* and the edges in the graph represent the relation between those nodes. Entailment depends upon the amount of semantic content of the given hypothesis present in the text.

As shown in fig. 2, (Haghighi et al., 2005), an example parse tree for the sentence "Bezos established company". In this sentence, there are three nodes: *Bezos and company as noun* and *established as verb*. In this manner, the graph of both text and hypothesis are drawn and the similarity is calculated in both the graphs to reach to a final conclusion.

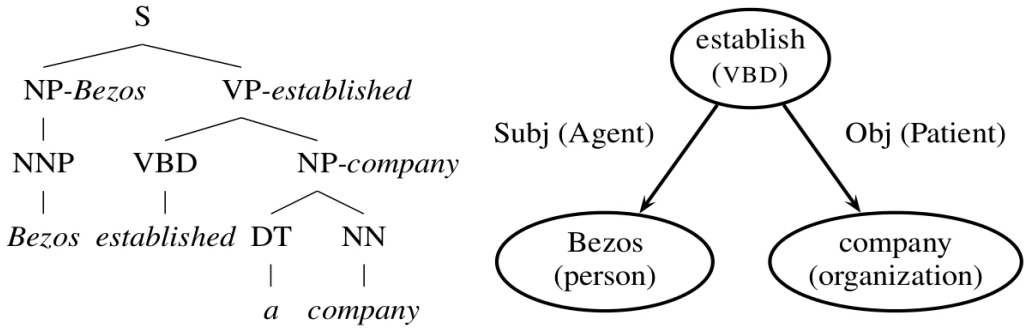

Figure 2: An example parse tree and its dependency graph

The first question that arises: *Why there is a need for dependencies?*, consider an example:

Consider the QA system which is searching for the answer of the question:

When was IIT Bombay established?

Without utilizing syntax, one can get an answer from the sentence:

CSE, IIT Bombay was established in 1973.

So, it is very important to match relationships along with the words in the pair of sentences.

Dependency graph is vital for the graphical representation of a sentence but there are certain concepts which should be resolved while dealing with it:

- Collapse Collocations and Named-Entities

  e.g., Generally a person name is more than one word, so it should be collapsed into a single node

- Dependency Folding

  e.g., Modifiers can become labels connecting the modifiers governor and dependent directly

- Coreference Links

  e.g., John is a hard-working boy, he will top the college. In this sentence, 'he' is referring to 'John', so it should be resolved.

4.7 Determining Entailment

Let, T be a Text graph, H be a Hypothesis graph and M(Matching) is a mapping which is used to map the vertices of T to vertices of H. For a vertex v in H, let the mapping of this vertex in T is $M(v)$. Similar to Statistical Machine Translation(SMT), some nodes may map to $NULL$ if necessary. Let the cost of matching M is $Cost(M)$. Suppose, X be the set of all matchings. Hencs, the cost of matching from H to T is:

$$MatchCost(H,T) = \min_{M \in X} Cost(M)$$

Let, there is a model $VertexSub(v, M(v))$, for substituting the vertex v in H to $M(v)$ in T and it gives a cost of [0,1]. Then,

$$VertexCost(M) = \frac{\sum\limits_{v \in H_v} w(v) VertexSub(v, M(v))}{Z}$$

$w(v)$ represents the weight or relative importance for vertex v and $Z = \sum\limits_{v \in H_v} w(v)$ is normalization constant.

Now, suppose there is a model $PathSub(e, \emptyset_M(e))$ for determining the "cost" of substituting a direct relation $e \in H_E$ for its counterpart, $\emptyset_M(e)$ under the matching.

So, computing the edge(relation) cost:

$$RelationCost(M) = \frac{\sum\limits_{e \in H_E} w(e) PathSub(e, \emptyset_M(e))}{Z}$$

where, $Z = \sum\limits_{e \in H_e} w(e)$ is normalization constant.

Final matching cost is given by a convex mixture cost:

$$Cost(M) = \alpha VertexCost(M) + (1 - \alpha) RelationCost(M)$$

5 Role of Knowledge Sources for TE

Knowledge plays an important role in any Artificial Intelligence problem, specially Textual Entailment. Success of the entailment system heavily depends on the background knowledge. Background knowledge includes facts, conventions, peculiar language features such as certain metaphors, idioms, proverbs, common beliefs *etc.* For example, when we say **kicking the bucket** we don't mean someone actually hitting the bucket with a leg. The background knowledge, together with the text should entail the hypothesis. If we represent the meanings of natural language expressions by logical formulas, for example in first order logic, we may think of textual entailment in terms of logical entailment. If the logical meaning representations of T and H are ϕ_T and ϕ_H, B is a knowledge base that contains all the background knowledge as first order formulas. Following example demonstrates the need for knowledge:

T: *John visited Mumbai.*

H: *John visited India.*

Here, without the knowledge that Mumbai is situated in India, it is not possible the recognize the entailment. This information must be the part of B. However, background alone should not entail the text. As per our notation, $B \vDash \phi_H$ does not imply $(\phi_T \Lambda B) \vDash \phi_H$. An example for this case would be:

T: *McDonald sell Happy Meals.*

H: *McDonald sell burgers.*

B: *..., McDonald is a fast food joint, ..., all fast food joints sell burgers,...*

Here, B alone entails H. The text T doesn't provide any information. Entailment will not be considered valid if text does not play any role in inferring hypothesis.

Apart from the general world knowledge (also termed as common sense), discourse often provides valuable knowledge that assists the entailment decision. Certain approaches try to extract the common beliefs (called as discourse) from the text and determine the entailment of H by the extracted discourse. (Hickl, 2008)

6 Application of TE in MT Evaluation

There are many applications of "text entailment". One of them is in the area of "Machine Translation Evaluation". This section focusses on this topic and

gives the whole idea about how TE can be beneficial while evaluating the MT output.

Terminology

Reference Sentence: This is the sentence which is the actual translation done by human for a given source language.
Candidate Sentence: This is the sentence which is generated by the machine translation system.

What is MT Evaluation

Machine Translation evaluation is the term that is used to judge the quality of the candidate sentence with respect to the given reference sentence. In the coming sections, we would see the work done in the field of MT evaluation. There are two kinds of MT evaluation:

Human Evaluation

This is the kind of evaluation which is done by the native speakers of the target language. It measures the output sentence according to adequacy and fluency.
Adequacy: Does the output convey the same meaning as the input sentence?
Fluency: Is the output good fluent according to the target language? This involves grammatical correctness of the sentence.

Automatic Evaluation

This is the evaluation which is done by some metric. This metric is responsible for the automatic judgement of the quality of the translation. Since human evaluation is costly and time consuming so there arised the need of automatic evaluation of MT output. In next section, we would discuss some of the popular automatic metrics.

7 Metrics for MT Evaluation

There exist many metric for MT evaluation such as: *BLEU*, *METEOR*, *TER* *etc.* In this section, the idea of some of the popular metrics is presented.

7.1 BLEU

BLEU (Bilingual Evaluation Understudy) is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another. Quality is considered to be the correspondence between a machine's output and that of a human. The central idea of BLEU is: "the closer a machine translation is to a professional human translation, the better it is". BLEU was one of the first metrics to achieve a high correlation with human judgements of quality, and remains one of the most popular automated and inexpensive metrics.[4]

BLEU score is evaluated in the range of 0 to 1. 0 indicates the low quality of translation and 1 indicates the best quality of translation with respect to the reference translation. BLEU score is the combination of *modified n-gram precision* and *brevity penalty* and is calculated as follow:

$$p_n = \frac{\sum_{c \in Candidates} \sum_{n-gram \in c} Count_{clip}(n-gram)}{\sum_{c' \in Candidates} \sum_{n-gram' \in c} Count(n-gram)} \tag{2}$$

$$BP = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c \geq r \end{cases} \tag{3}$$

$$BLEU = BP * exp\left(\sum_{n=1}^{N} w_n log p_n \right) \tag{4}$$

where, eq. (2) and (3) are the mathematical formulation for modified n-gram precision and brevity penalty *respectively*. Eq. (4) states the formula of BLEU score.

7.2 METEOR

METEOR (Metric for Evaluation of Translation with Explicit ORdering) is based on the harmonic mean of unigram precision and recall, with recall weighted higher than precision. It takes care of some linguistic features such as: synonymy, stemming and the exact word matching. It first creates an alignment between the candidate and reference translation. Here, alignment is referring to the mapping of *unigrams* of both the sentences.

A mapping can be thought of as a line between a unigram in one string, and a unigram in another string. The constraints are as follows; every unigram in the candidate translation must map to zero or one unigram in the reference. Mappings are selected to produce an alignment as defined above.

[4]http://en.wikipedia.org/

If there are two alignments with the same number of mappings, the alignment is chosen with the fewest crosses, that is, with fewer intersections of two mappings.[5] After the alignment of unigrams, unigram precision (P) and unigram recall (R) are calculated as follows:

$$P = \frac{m}{w_t} \tag{5}$$

$$R = \frac{m}{w_r} \tag{6}$$

where, m is the matching number of unigrams in the candidate translation that are present in the reference translation, and w_t is the total number of unigrams in the candidate translation. w_r is the number of unigrams in the reference translation.

After the calculation of precision and recall, harmonic mean of the two is computed as follows (in which, recall is weighted more in comparison to precision):

$$F_{mean} = \frac{10PR}{R + 9P} \tag{7}$$

The measures that have been computed above only take care of single words but what about the larger segments that appear in both the reference and the candidate sentence. For handling this, longer n-gram matches are used to compute a penalty p for the alignment. The more mappings there are that are not adjacent in the reference and the candidate sentence, the higher the penalty will be.

In order to compute this penalty, unigrams are grouped into the fewest possible chunks, where a chunk is defined as a set of unigrams that are adjacent in the candidate as well as in the reference. The longer the adjacent mappings between the candidate and the reference, the fewer chunks there are. A translation that is identical to the reference will give just one chunk. The penalty p is computed as follows:

$$p = 0.5\left(\frac{c}{u_m}\right)^3 \tag{8}$$

where, c is the number of chunks, and u_m is the number of unigrams that have been mapped. The final METEOR score for a segment is calculated as follows:

$$M = F_{mean}(1 - p) \tag{9}$$

[5]http://en.wikipedia.org

For calculating the score over a whole corpus, or collection of segments, the aggregate values for P, R and p are calculated and then combined using the same formula as eq. (9).

7.3 TER

TER (Translation Error Rate) measures the number of edits required to change a system output into the given reference sentence. TER can be formulated as follows:

$$TER = \frac{\text{No. of Edits required}}{\text{No. of words in the Reference Sentence}} \tag{10}$$

It takes care of four kinds of operations: shift, insert, delete and substitution.

8 Related Work in MT Evaluation

First evaluation of machine translation was done by (ALPAC, 1966). In this, they asked evaluators to adjudge intelligibility and fidelity of the translations. (Slype, 1979) developed the SYSTRAN system in which instead of looking for correctness of the translation, he adjudged SYSTRAN for acceptability. Here the evaluators were asked if translation A is better than translation B. The prime objective of this evaluation was to distinguish between correct sentences from incorrect ones. This evaluation developed a tendency to give a measure to check the output of the system, and also found the cost of post editing the incorrect translations.

In 1992, DARPA compared MT system outputs using a comprehension test for intelligibility and a quality test for fidelity (White, 1993). They took passages from various texts as source for translation. They analyzed that this was a very complex and highly expensive method of evaluation, thus in subsequent years they simplified comprehension test. Moreover, the quality test was replaced with adequacy and fluency tests which were assessed on a scale of 1-5.

(Church and Hovy, 1993) looked at measuring informativeness of the translation. They directly compared the MT systems onto the results of comprehension tests where human evaluators were asked to read MT outputs and then answered certain multiple choice questions. Their argument was that if the translations can capture the information correctly then the user must be able to answer certain questions based on this information.

Score	Description
4	Exactly the same meaning
3	Almost the same meaning
2	Partially the same meaning and no new information
1	Partially the same meaning but misleading information is introduced
0	Totally different meaning

(Papineni et al., 2002) proposed BLEU as an automatic MT evaluation metric which is based on the n-gram matching of the reference and candidate sentences. This is still considered as the most reliable metric and used widely in the MT community for the determination of the translation quality. BLEU averages the precision for unigram, bigram and up to 4-gram and applies a length penalty if the generated sentence is shorter than the best matching (in length) reference translation.

Studies such as (Callison-Burch et al., 2006) and (Zhang et al., 2004) have shown that BLEU and related n-gram-based scores have a number of problems including (1) BLEU is unreliable at the segment-level due to data sparsity (2) BLEU scores are biased towards statistical MT systems (3) BLEU does not always reflect the translation quality differences between MT and human translations.

(Vanni and Miller, 2002) used clarity as a measure of ascertaining MT quality. They asked the human evaluators to score MT outputs on the basis of the clarity of the translation on a scale of 0-3.

Alternative approaches have been designed to address problems with BLEU. The NIST metric (Church and Hovy, 2002) is derived from the BLEU evaluation criterion but differs in one fundamental aspect: instead of n-gram precision, the information gain from each n-gram is taken into account. The idea behind this is to give more credit if a system gets an n-gram match that is difficult, but to give less credit for an n-gram match which is easy. TER (Snover et al., 2006) tries to improve the hypothesis/reference matching process based on the edit-distance and METEOR (Banerjee and Lavie, 2005) considers linguistic evidence, mostly lexical similarity, for more intelligent matching. (Liu and Gildea, 2005), (Owczarzak et al., 2007), and (Zhang et al., 2004) use syntactic overlap to calculate the similarity between the hypothesis and the reference. Pado et al. (2009) proposes a metric that evaluates MT output **based on a rich set of textual entailment features** such as lexical-semantic compatibility and argument structure.

MT evaluation is gaining its popularity that can be seen from the recent works. (Doherty et al., 2010) proposed a work of MT evaluation using eye tracking. In this work, they analyzed that gaze time and fixation time count is more with bad translation than for good translation.

9 Role of Entailment in MT Evaluation

We discussed various aspects of MT evaluation in last sections. Usage of *entailment* phenomenon provides the well formedness of the output sentence generated by the translation system. During the generation of the score from entailment system, dependency parsers are used to generate the dependencies of the sentences and if the sentence is well formed then only the dependencies would be reliable. It may be the possibility that while generation of the output by a translation system, some words from the source sentence can appear in the output. This will decrease the quality of the output sentence. Hence the parser would not be able to handle this kind of situation and the genrated dependencies would not be appropriate. After that when the entailment procedure will be applied, then the generated score for such kind of sentences will be low compared to the well formed sentences.

Text Entailment is the kind of inferring the meaning. So, while applying it in MT evaluation, this phenomenon should be applied in both directions. Because, *reference* should entail the meaning of *candidate* and *vice-versa*.

References

ALPAC (1966). Languages and machines: Computers in translation and linguistics (technical report).

Banerjee, S. and Lavie, A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments.

Callison-Burch, C., Osborne, M., and Koehn, P. (2006). Re-evaluation the role of bleu in machine translation research.

Church, K. W. and Hovy, E. H. (1993). Good applications for crummy machine translation.

Church, K. W. and Hovy, E. H. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics, nist.

Dagan, I., Glickman, O., and Magnini, B. (2005). The pascal recognising textual entailment challenge.

Doherty, S., O'Brien, S., and Carl, M. (March, 2010). Eye tracking as an automatic mt evaluation technique.

Glickman, O. and Dagan, I. (2005). *A Probabilistic Setting and Lexical Cooccurrence Model for Textual Entailment*. EMSEE '05. Association for Computational Linguistics.

Haghighi, A. D., Ng, A. Y., and Manning, C. D. (2005). Robust textual inference via graph matching. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 387–394. Association for Computational Linguistics.

Hickl, A. (2008). Using discourse commitments to recognize textual entailment. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08.

Johan Bos, K. M. (2005). Recognising textual entailment with logical inference. In *In EMNLP-05*, pages 628–635.

Liu, D. and Gildea, D. (2005). Syntactic features for evaluation of machine translation.

MacCartney, B. (June, 2009). *Natural Language Inference, Ph.D. thesis*. Stanford University.

Owczarzak, K., van Genabith, J., and Way, A. (2007). Evaluating machine translation with lfg dependencies.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: A method for automatic evaluation of machine translation.

Prodromos Malakasiotis, I. A. (2007). *Learning Textual Entailment using SVMs and String Similarity Measures*. Association for Computational Linguistics.

Slype, G. (1979). Systran: evaluation of the 1978 version of systran english-french automatic system of the commission of the european communities.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation.

Vanni, M. and Miller, K. (2002). Scaling the isle framework: Use of existing corpus resources for validation of mt evaluation metrics across languages.

White, J. (1993). Evaluation of machine translation.

Zhang, Y., Vogel, S., and Waibel, A. (2004). Interpreting bleu/nist scores: how much improvement do we need to have a better system?