Using Example-Based MT to Support Statistical MT when Translating Homogeneous Data in a Resource-Poor Setting

Sandipan Dandapat, Sara Morrissey, Andy Way, Mikel L. Forcada

CNGL, School of Computing, DCU

Introduction

- Over the past two decades, statistical MT (SMT) has shown very promising results
  - Requires reasonably good amount of parallel corpora
- A large number of languages suffer from the scarcity of large parallel corpora
  - Indic languages, Sign languages etc.
- Some studies have shown SMT approaches have yielded low translation quality for these poorly resourced languages (Islam et al, 2010; Khalilov et al., 2010).

Introduction

- Domain-specific translation to tackle the issue of scarce resources
  - Very low accuracy within SMT framework for homogeneous domain (Dandapat et. al., 2010)
- Can example-based MT (EBMT) techniques help?
  - EBMT approach can be developed using a limited example base (Somers, 2003)
  - EBMT system works well when training and test data are quite **close** in nature (Marcu, 2001)

Our Attempt

- We adopt two different EBMT approaches for translating homogeneous data in a resource-poor setting

I. A compiled approach to EBMT

- Produces **translation templates** during the training stage (Cicekli and Güvenir, 2001)

II. A novel way of integrating TM into an EBMT system

- Using a subsentential TM (extracted using an SMT system) in the alignment and recombination stages of an EBMT system

Structure of the Corpus

- The size and type of corpora is important for adopting a particular data-driven approach to MT
- We use the IWSLT 2009 English–Turkish corpus to deal with less-resourced homogeneous data.
  - The training data is quite small (20k parallel sentences)
  - Corpus is comprised of very similar domain-specific sentences

1.	(a)	Have you ever *seen a* Japanese *movie* ?	2.	(a)	I'd like to *see that camera* on the *shelf* .
	(b)	Have you ever *tried* Japanese *food* ?		(b)	I'd like to *have it parted* on the *left* .

Approach I

- Generalized translation-template-based EBMT
  - **Learning phase:** learn templates from sentence-aligned bitext
  - **Decoding phase:** translate new sentences using the translation templates

Generalized translation-template-based EBMT

- **Learning phase** - learns templates from bitext by studying similarities and differences between two example pairs (Cicekli and Güvenir, 2001:p. 58)

I will drink orange juice → *portakal suyu içeceğim*

I will drink coffee → *kahve içeceğim*

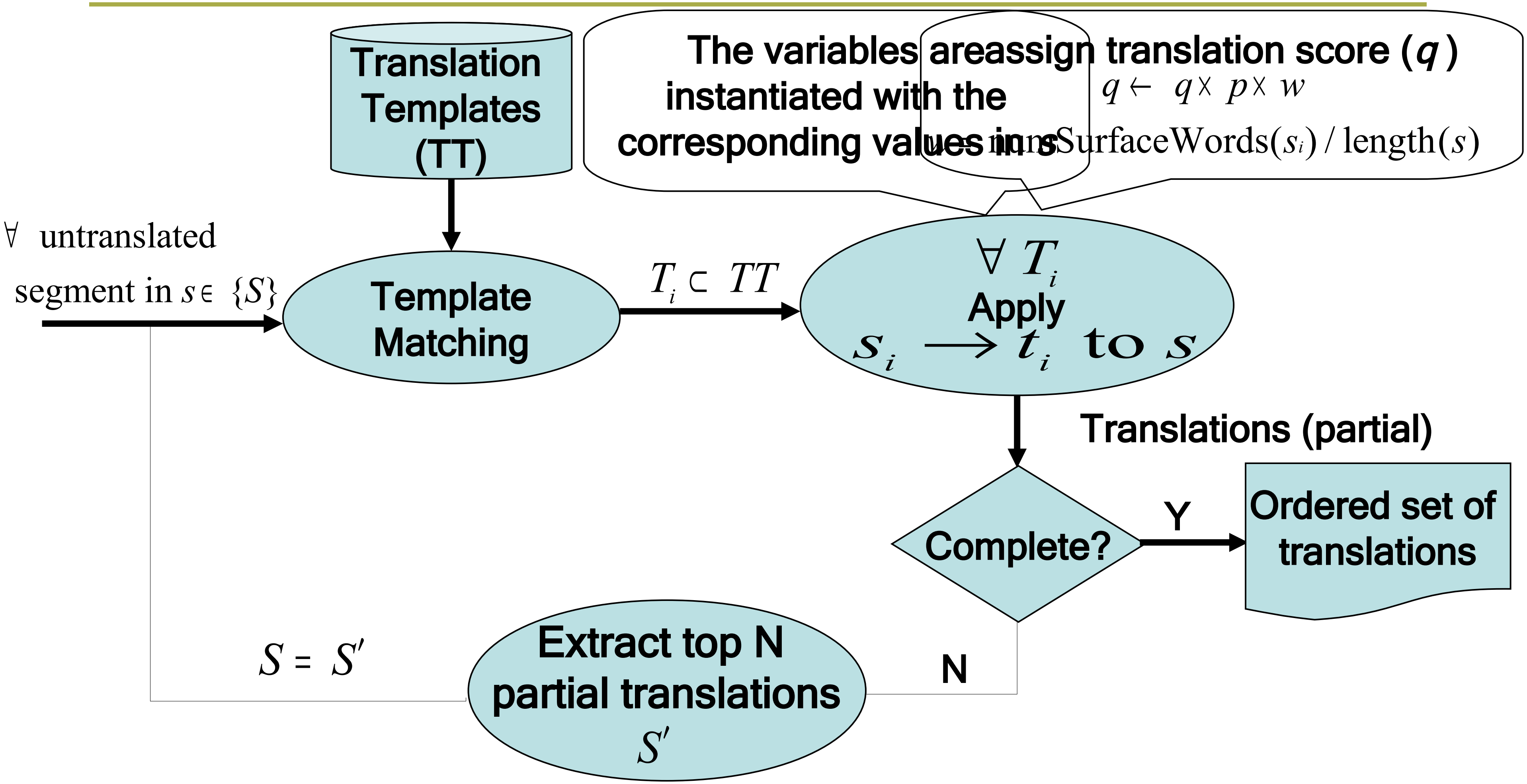
I will drink → *içeceğim*	*I will drink* X^S → X^T *içeceğim*
coffee → *kahve*	X^S *orange juice* → *portakal suyu* X^T
orange juice → *portakal suyu*	X^S *coffee* → *kahve* X^T

We assign a probabilistic score (p) to each translation template $T_i : s_i \rightarrow t_i$

$$p(t_i \mid s_i) = count(s_i \rightarrow t_i) / count(s_i)$$

X^S *orange juice* → *portakal suyu* X^T 0.33(p)

Decoding



Translation Templates (TT)
The variables are instantiated with the corresponding values in s
assign translation score (q)
$q \leftarrow q \times p \times w$
w = numSurfaceWords(s_i) / length(s)
$\forall$ untranslated segment in $s \in \{S\}$
Template Matching
$T_i \subset TT$
$\forall T_i$ Apply $s_i \rightarrow t_i$ to s
Translations (partial)
Complete?
Y
Ordered set of translations
N
Extract top N partial translations S'
$S = S'$

Approach II

- EBMT using subsentential TM
  - **Matching** – finds the closest match with the input sentence
  - **Alignment** – finds translation of the desired segments
  - **Recombination** – combines the translations of the desired segments

Building a Subsentential TM

- We build an auxiliary subsentential TM automatically from the English–Turkish small training corpus
- We use Moses to automatically build this TM
  - Aligned phrase pairs from the Moses phrase table
  - Aligned word pairs based on GIZA++

Entries in TM from Moses phrase table	
i don't like it	{"sevmedim", "bunu sevmedim"}
i can't sleep well.	{"iyi uyuyamıyorum ."}

Entries in TM from word-alignment	
helps	{"vücudun", "yardım", "eder"}
coffees	{"kahve"}

- We keep all target equivalents sorted according to phrase translation probability

Matching

- We find *the closest sentence* (s_c) from the example base for *the input sentence* (s) to be translated

$$s_c = \arg\max_i \ \mathrm{score}(s, s_i)$$

- Edit distance metric to find this closest match sentence

$$\mathrm{score}(s, s_i) = 1 - \mathrm{ED}(s, s_i) / \max(|s|, |s_i|)$$

s :	*i'd like a present for my mother .*
s_c :	*i'd like a shampoo for greasy hair .*

- We consider the associated translation (t_c) of s_c to build the skeleton for the translation of the input sentence s

t_c :	*yağlı*	*saçlar*	*için*	*bir*	*şampuan*	*istiyorum .*
	GREASY	HAIR	FOR	ONE	SHAMPOO	I'D-LIKE .

Alignment

- We extract the translation of the non-matching fragments of the input sentence (s)
- To do this, we align three sentences – the input (s), the closest source-side match (s_c) and its target equivalent (t_c)

1. Mark the mismatched portion between input sentence (s) and the closest source-side match (s_c) using edit distance

s : i'd like a <*present*> for <*my mother*> .

s_c : i'd like a <*shampoo*> for <*greasy hair*> .

Alignment

- We extract the translation of the non-matching fragments of the input sentence (s)
- To do this, we align three sentences – the input (s), the closest source-side match (s_c) and its target equivalent (t_c)

2. We align the mismatched portion of s_c with its associated translation t_c using our TM

s : i'd like a <*present*> for <*my mother*> .

s_c : i'd like a <*shampoo*> for <*greasy hair*> .

t_c : <1:*yağlı saçlar*> için bir <0:*şampuan*> istiyorum .

- The numbers in angle brackets keep track of the order of the appropriate fragments

Recombination

- Substitute, add or delete segments from the input sentence (s) with the translation skeleton (t_c).

s : i'd like a <*present*> for <*my mother*> .

s_c : i'd like a <*shampoo*> for <*greasy hair*> .

t_c : <1:*yağlı saçlar*> için bir <0:*şampuan*> istiyorum .

t(*my mother*) = ?   t(*present*) = ?

➢ <1: t(*my mother*)> için bir <0: t(*present*)> istiyorum.

- We estimate the t(·) from our subsentential TM.
  - Recursively translating the longest possible matched segment in TM

Experiments

- Baseline SMT (using Moses)
- **GEBMT** – baseline experiment with generalized translation template-based EBMT
- **EBMT** – based only on the matching step. Considering closest match target (t_c) as the output
- **EBMT$_{TM}$** - after obtaining the translation skeleton, unmatched segments are translated using subsentential TM
- English–Turkish data used for experiments
  - Training Data - 20k sentences (IWSLT’09 training data)
  - Test Data – 414 sentences (IWSLT’09 devset)

Combining the Systems with SMT

- EBMT systems (GEBMT and $EBMT_{TM}$) sometimes produce correct solutions where SMT fails and vice-versa
- We combine GEBMT and SMT based on the translation score (*q*) for an input test sentence (*s*)
  - If the value of *q* is greater than some threshold we rely on GEBMT(*s*) otherwise we take the output from SMT(*s*)
- We call this **$GEBMT_{score > x}$ + SMT**
- We combine $EBMT_{TM}$ and SMT (**$EBMT_{TM}$ + SMT**) based on two features
  - Fuzzy match score (FMS)
  - The equality in number of mismatched segments in s, s_c and t_c (EqUS)
- Rely on $EBMT_{TM}$ output depending on these two features

Results

Accuracy obtained with GEBMT system using very small data

Accuracy obtained with GEBMT system with little more data

System	BLEU(%)
Training Data: 1242 sentences	
SMT	7.63
GEBMT	6.80
$GEBMT_{score>0.3}$ +SMT	**7.96**
Training Data: 2184 sentences	
SMT	10.72
GEBMT	07.21
$GEBMT_{score>0.9}$ +SMT	**10.83**
$GEBMT_{score>0.8}$ +SMT	**10.99**
$GEBMT_{score>0.7}$ +SMT	10.76
GEBMT [illegible] +SMT	10.55

Results

Accuracy obtained with $EBMT_{TM}$ system

Accuracy obtained with $EBMT_{TM}$ + SMT system

System		BLEU(%)
Training Data: 19,922 sentences		
SMT		**23.59**
EBMT		15.60
$EBMT_{TM}$		20.08
System: $EBMT_{TM}$ + SMT		
Condition	**time/percentage $EBMT_{TM}$ used**	**BLEU(%)**
FMS >0.85	35 (8.5%)	24.22
FMS >0.8	114 (27.5%)	23.99
FMS >0.7	197 (47.6%)	22.74
FMS >0.85 & EqUS	**24 (5.8%)**	**24.41**
FMS >0.8 & EqUS	76 (18.4%)	24.19
FMS >0.7 & EqUS	127 (30.7%)	24.08

Assessment of Error Types

- Incorrect alignment in matching phase
  - Due to erroneous TUs in the subsentential TM

s: i have a terrible <*headache*> .

s_c: i have a terrible <*cough*> .

t_c: berbat bir öksürüğüm var .

cough → {"öksürük","öksürük tedavisi için"} in TM

t': berbat bir öksürüğüm var baş ağrısı.

- Incorrect translation produced during decoding
  - Mostly when falling back to word-based translation

- Incorrect morpho-syntactic alignment

s: do you have a japanese <*guidebook*> ?

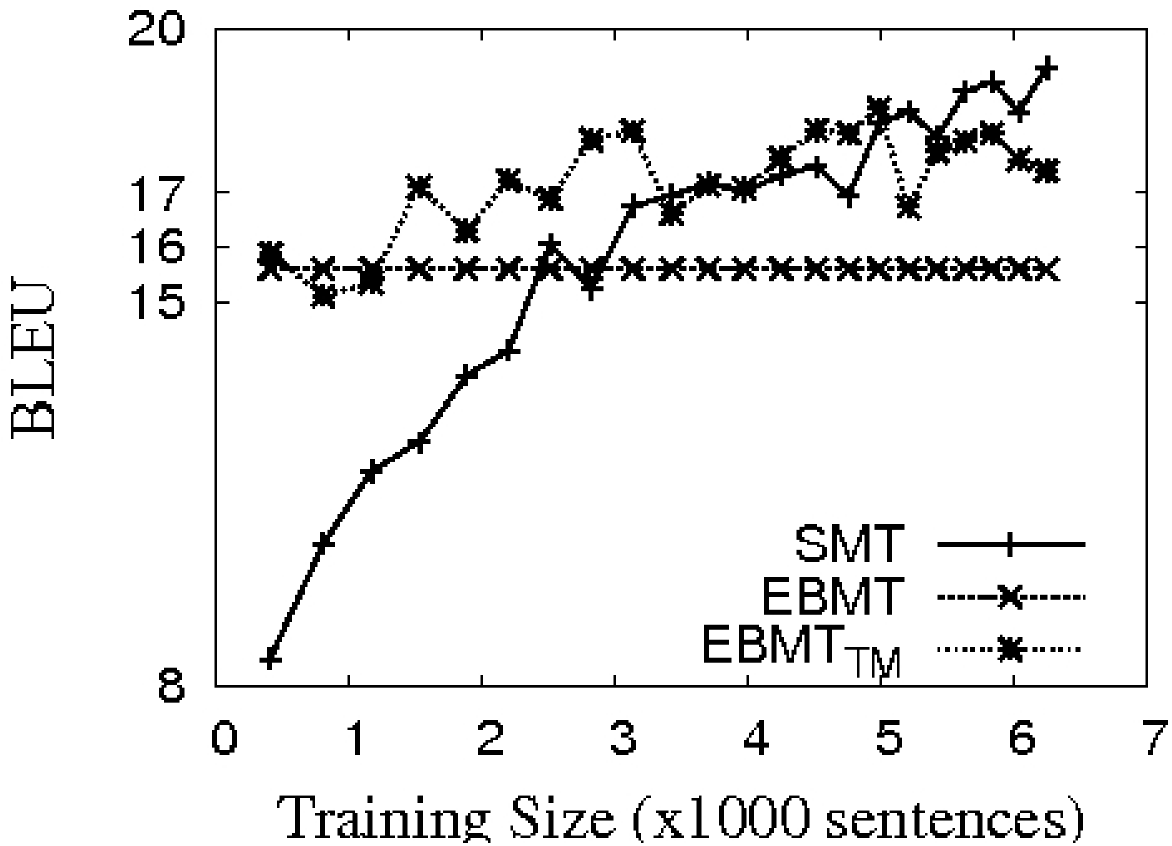
s_c: do you have a japanese <*magazine*> ?

t_c: japonca bir <0:*derginiz*> var mı ?

t': japonca bir rehber kitap var mı ?

Observations

- Effect of training data size in $EBMT_{TM}$ system



Observations

- GEBMT system has lower accuracy on its own compared to baseline SMT
- Combining GEBMT with SMT has some improvement over SMT
  - relative BLEU improvement of 4.3% with 1242 sentences; less (2.5% relative BLEU) with 2184 sentences

- $EBMT_{TM}$ system has higher score than baseline when the amount of data is small
- With increased data size, SMT performs better compared to $EBMT_{TM}$ system
- Combing $EBMT_{TM}$ and SMT using FMS and EqUS shows improvement over the baseline SMT

Conclusion

- EBMT works better for certain sentences when the amount of available resources is limited
- Combining EBMT and SMT may be expected to yield a higher score than an individual system
- Integration of subsentential TM with EBMT improves translation quality

Future Work

- In order to test the scalability, we plan to use larger training and test data
- We intend to find more sophisticated features (other than FMS and EqUS) to trigger the use of EBMT system

Thank You

Questions?