Interactive-Predictive Speech-Enabled Computer-Assisted Translation

Shahram Khadivi, Zeinab Vakil

Human Language Technology Lab, Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran
{Khadivi, Z.Vakil}@aut.ac.ir

Abstract

In this paper, we study the incorporation of statistical machine translation models to automatic speech recognition models in the framework of computer-assisted translation. The system is given a source language text to be translated and it shows the source text to the human translator to translate it orally. The system captures the user speech which is the dictation of the target language sentence. Then, the human translator uses an interactive-predictive process to correct the system generated errors. We show the efficiency of this method by higher human productivity gain compared to the baseline systems: pure ASR system and integrated ASR and MT systems.

1. Introduction

Nowadays, with the expansion of global communications, the need for the translation has become a basic and important requirement, especially for international institutions and news agencies. Consider the following example to illustrate the importance of the translation in today world. In 2003, after the enlargement of the European Union, with a population of 453 million, the cost of the translation at all institutions, once translators are operating at full speed, was estimated at 807 M€ per year.

Recently, significant improvements have been achieved in statistical machine translation (MT), but still even the best machine translation technology is far from replacing or even competing with human translators. In order to achieve high quality translations, translated texts by these systems need to be reviewed and corrected by a human translator.

Another way to increase the productivity of the translation process is computer-assisted translation (CAT) system. In a CAT system, the human translator begins to type the translation of a given source text; by typing each character the MT system interactively offers the choices to enhance and complete the translation. Human translator may continue typing or accept the whole completion or part of it.

Interactive machine translation (IMT), first appeared as part of Kay's MIND system [1], where the user's role was to help with source-text disambiguation by answering questions about word sense, pronominal reference, prepositional-phrase attachment, etc. Later work on IMT, eg [2,3,4], has followed in this vein, concentrating on improving the question/answer process by having less questions, more friendly ones, etc. Despite progress in these endeavors, the question/answer process remained in the systems of this sort. Finally these systems are only used where the cost of manually producing a translation is high enough to justify the extra effort, for example when the user's knowledge of the target language may be limited or non-existent, or when there are multiple target languages. With introducing TransType project by [5], a major change in how the user interacts with the machine had occurred. In such an environment, human translators interact with a translation system that acts as an assistance tool and dynamically provides a list of translations (suffixes) which complete the part of the source sentence already translated (prefix). Also from 1997 to 2004, most of the given papers related to the various versions of the TransType project such as [6,7,8,9].

Also one desired feature of a computer-assisted translation system is to provide an environment to accept the translator's target language speech signal to speed up the translation process; since professional translators can translate a given text faster by dictation rather than directly typing the translation [10]. In such a system, two sources of information are available to recognize the speech input; the target language speech and the given source language text. The target language speech is just a human-produced translation of the source language text. Machine translation models are used only to take into account the source text in order to increase the speech recognition accuracy. The overall schematic of automatic text dictation in computer-assisted translation is depicted in Figure 1.

The idea of incorporating statistical machine translation and speech recognition models was independently initiated about one decade ago by two groups: researchers at the IBM Thomas J. Watson Research Center [10] and researchers involved in the TransTalk project [11] and [12].

In [10], the authors described the statistical speech recognition models and statistical translation models. Then, they proposed a method for combining those models, but they did not report any recognition or translation results. Instead, they just reported the perplexity reduction when the translation models were combined to recognition models.

In the TransTalk project [11] and [12], the authors reported three different combination methods between translation and recognition models. The first method was capable only of isolated word recognition. In the second method, the speech recognition system generates a list of the most probable word sequence hypotheses. Then the statistical translation models rescore them and select the best word sequence hypothesis. The idea behind the third method was the dynamic vocabulary for a speech recognition system which translation models generated for each source language sentence. The best recognition results have been achieved with the second method, while the third method was faster. The authors have shown the promising results of combining the translation models to speech recognition models. However, they neither described the details of the utilized translation model nor studied the impact of different translation models. Also recently, some researcher in [13,14,15,16,17] have studied the integration of ASR and MT models but in the any of these works haven't been used from interactive framework. For the first time, in this paper, we enter interactive form into a speech enabled CAT and create a Speech-Enabled Interactive CAT. In this new system, the human translator uses an interactive-predictive process to correct the system generated errors.

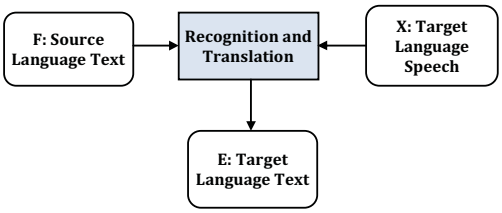

Figure 1: Schematic of automatic text dictation in computer-assisted translation

2. Models of interactive-predictive speech-enabled CAT

In a speech-enabled interactive-predictive computer-assisted translation system, we are given a source language sentence $F = f_1 \dots f_j \dots f_J$, an acoustic signal $X = x_1 \dots x_t \dots x_T$ that is the speech of the target language sentence, and the correct translated part of the target language sentence (prefix) $E_p = e_1 \dots e_i$. Then, we generate the best complement for the target sentence prefix (suffix) $E_s = e_{i+1} \dots e_I$. Among all possible target language sentence suffixes, we will choose the sentence with the highest probability:

$$\hat{E}_s = \text{argmax}_{E_s}\{P(E_s, E_p, F, X)\} \quad (1)$$

$$= \text{argmax}_{E_s}\{P(E_p, E_s, F).P(X|E_p, E_s, F)\} \quad (2)$$

$$= \text{argmax}_{E_s}\{P(E_s, E_p).P(F|E_p, E_s).P(X|E_p, E_s)\} \quad (3)$$

$$= \text{argmax}_{E_s}\{P(E_s|E_p).P(F|E_p, E_s).P(X|E_p, E_s)\} \quad (4)$$

Equation 2 is simplified into Equation 3 by assuming that there is no direct dependence between X and F. The decomposition into three knowledge sources in Equation 4 allows an independent modelling of the target language model $P(E_s|E_p)$, the translation model $P(F|E_p, E_s)$ and the acoustic model $P(X|E_p, E_s)$.

The target language model describes the well-formedness of the target language sentence. The translation model links the source language sentence to the target language sentence. The acoustic model links the acoustic signal to the target language sentence. The argmax operation denotes the search problem, i.e. the generation of the output sentences in the target language by maximization all possible target language sentences. Another approach for modelling the posterior probability $P(E_s|E_p, F, X)$ is direct modelling by the use of a log-linear model. The direct posterior probability is given by:

$$P(E_s|E_p, F, X) = \frac{\exp[\sum_{m=1}^{M} \lambda_m h_m(E_s, E_p, F, X)]}{\sum_{\hat{E}_s} \exp[\sum_{m=1}^{M} \lambda_m h_m(\hat{E}_s, E_p, F, X)]} \quad (5)$$

This approach has been suggested by Papineni et al. in [18.19], for natural language understanding task; by Beyerlein in [20], for automatic speech recognition; and in [21] for statistical machine translation. The time-consuming renormalization in Equation 5 is not needed in the search. Therefore we obtain the following decision rule:

$$\hat{E}_s = \text{argmax}_{E_s} \sum_{m=1}^{M} \lambda_m h_m(E_s, E_p, F, X) \quad (6)$$

Each of the terms $h_m(E_s, E_p, F, X)$ denotes one of the various models which are involved in the recognition process. Each individual model is weighted by its model scaling factor λ_m. As there is no direct dependence between F and X, the $h_m(E_s, E_p, F, X)$ can be in one of these two forms: $h_m(E_s, E_p, X)$ and $h_m(E_s, E_p, F)$.

This approach is a generalization of Equation (6). The direct modeling has the advantage that additional models or feature functions can be easily integrated into the overall system. Based on Equation (4), the principal models which will contribute to the final system are the acoustic model, the language model, and the translation model(s). We may use one or more translation models in the final system. A set of possible translation models consists of *HMM*, *IBM-1*, *IBM-2*, *IBM-3*, *IBM-4*, *IBM-5*, and *Alignment Template* models, which will be described in Section 3. The details of utilized acoustic and language models will be explained in Section 4.

The model scaling factors λ_1^M in Equation 5 are trained according to the maximum entropy principle, e.g. using the GIS algorithm. Alternatively, one can train them with respect to the final recognition quality measured by the word error rate [22]. The development of an efficient search algorithm for integrating automatic speech recognition and statistical machine translation models is very complicated. Thus, in order to facilitate the implementation of the above log-linear model, we use the principle of *N*-best rescoring instead of implementing a new search algorithm. The *N*-best rescoring approach helps us to quickly examine many different dependencies and models for the combination of automatic speech recognition and statistical machine translation.

The recognition process is performed in two steps. In the first step, the baseline speech recognition system creates an *N*-best list of length *N* for every utterance *X* of the given corpus. In the second step, the translation models rescore every sentence pair (the entries in the *N*-best list with their corresponding source sentence). For each utterance, the decision about the best recognized sentence is made according to the recognition and the translation models. Then the implementation approach is very similar to the second method explained in [12].

3. Translation models

A key issue in modeling the translation model probability $P(F|E_p, E_s)$ is the question of how we define the correspondence between the words of the target sentence and the words of the source sentence. In typical cases, we can assume a sort of pairwise dependence by considering all word pairs (f_j, e_i) for a given sentence pair $(f_1^J; e_1^I)$. A family of such *alignment models* (IBM-1,..., IBM-5) was developed in [23]. Using the similar principles as in Hidden Markov models (HMM) for speech recognition, we re-write the translation probability by introducing the *hidden alignments* $\boldsymbol{\mathcal{A}}$ for each sentence pair $(f_1^J; e_1^I)$:

$$Pr(f_1^J|e_1^I) = \sum_{\boldsymbol{\mathcal{A}}} Pr(f_1^J, \boldsymbol{\mathcal{A}}|e_1^I) \quad (7)$$

IBM-1,2 and Hidden Markov Models. The first type of alignment models is virtually identical to HMMs and is based on a mapping $j \rightarrow i = a_j$, which assigns a source position j to a target position $i = a_j$. Using suitable modeling assumptions [22,23], we can decompose the probability $Pr(f_1^J, \boldsymbol{\mathcal{A}}|e_1^I)$ with $\boldsymbol{\mathcal{A}} = a_1^J$:

$$Pr(f_1^J, a_1^J|e_1^I) = p(J|I).\prod_{j=1}^{J}\left[p(a_j|a_{j-1}, I, J).p\left(f_j\middle|e_{a_j}\right)\right] \quad (8)$$

With the length model $p(J|I)$, the alignment model $p(i|i',I,J)$ and the lexicon model $p(f_j|e_i)$. The alignment models IBM-1 and IBM-2 are obtained in a similar way by allowing only zero-order dependencies.

IBM-3, 4 and 5 Models. For the generation of the target sentence, it is more appropriate to use the concept of inverted alignments which perform a mapping from a target position *i* to a set of source positions *j*, i.e. we consider mappings $\mathcal{B}$ of the form:

$$\mathcal{B}: i \rightarrow \mathcal{B}_i \subset \{1, \ldots, j, \ldots, J\} \quad (9)$$

with the constraint that each source position j is covered exactly once. Using such an alignment $\mathcal{A} = \mathcal{B}_1^I$ we re-write the probability $Pr(f_1^J, \mathcal{A}|e_1^I)$:

$$Pr(f_1^J, \mathcal{B}_1^I|e_1^I) = p(J|I) \cdot \prod_{i=1}^{I}\left[p(\mathcal{B}_i|\mathcal{B}_1^{i-1}) \cdot \prod_{j \in \mathcal{B}_i} p(f_j|e_i)\right] \quad (10)$$

By making suitable assumptions, in particular first-order dependencies for the inverted alignment model $p(\mathcal{B}_i|\mathcal{B}_1^{i-1})$, we arrive at what is more or less equivalent to the alignment models IBM-3, 4 and 5 [24].

Alignment Template Model. In all the above models, the single words are taken into account. In [25,26], the authors showed significant improvement in translation quality by modeling *word groups* rather than *single words* in both the alignment and lexicon models. This method is known as the *alignment template* (AT) approach.

3.1. Training

The unknown parameters of the alignment and lexicon models are estimated from a corpus of bilingual sentence pairs. The training criterion is the maximum likelihood criterion. As usual, the training algorithms can guarantee only local convergence. In order to mitigate the problems with poor local optima, we apply the following strategy [23]. The training procedure is started with a simple model for which the problem of local optima does not occur or is not critical. The parameters of the simple model are then used to initialize the training procedure of a more complex model, in such a way that a series of models with increasing complexity can be trained. To train the above models except for the alignment template model, we use the GIZA++ software [24]. The alignment template model training scheme, and also the description of our translation system which is based on the alignment template approach is explained in [26].

4. Speech recognition system

The speech recognition system is trained on a large vocabulary, namely the European Parliament Plenary Sessions (EPPS) corpus. The corpus consists of: 67k training-sentences (87.5h) from 154 speakers. The other statistics of the speech recognition train corpus are shown in Table 1.

4.1. Experimental results

We rescore the ASR *N*-best lists with the standard HMM [27] and IBM [23] MT models. Then we use each the *N*-best list as *N*-best hypotheses in order to provide target suffixes for the CAT system.

Table 1: Statistics of the speech recognition train corpus.

	EPPS
Language	English
Acoustic data [h]	87.5
# Running words	705 K
Vocabulary size	58 K
# Segments	67 K
# Speaker	154

The size of the development and evaluation sets *N*-best lists is sufficiently large to achieve almost the best possible results. On average 1738 hypotheses per each source sentence are extracted from the ASR word graphs. The ASR and MT integration experiments are carried out on a large vocabulary task which is the Spanish–English parliamentary speech translation (EPPS). The corpus statistics is shown in Table 2.
To determine the performance of the speech-enabled interactive-predictive CAT system, we simulate a human translator who uses this system. The simulated human knows the correct translation and selects all or part of a suggested suffix whenever this suffix matches fully or partially with the correct translation. If suggested suffix doesn't match with the reference translation, simulated human will more complete the prefix, character by character, until whole or part of a suggested suffix matches with the reference translation.
See Figure 2 for the pseudo-code of the algorithm that simulates a human, matches prefix in the N-best lists and calculates the measure of user efforts.

Table 2: Statistics of the Spanish-English (EPPS) corpus.

		EPPS	
		Spanish	English
Train	Sentences	1 167 627	
	Running words	35.3 M	33.9 M
	Vocabulary size	159 080	110 636
	Singletons	63 045	46 121
Dev	Sentences	1 750	
	Running words	22 174	23 429
	OOVs	64	83
Test	Sentences	792	
	Running words	19 081	19 306
	OOVs	43	45

4.2. Evaluation metrics

In order to measure the performance of our CAT system, we need to determine quantity of effort the human translator for translating a sentence in the absence and presence of the CAT system. For this purpose, we use the summation of the keystroke ratio (KSR) and mouse-action ratio (MAR) which in follow are described.
KSR (Key-stroke ratio): The KSR is the number of key-strokes required to produce the single reference translation using the interactive machine translation system divided by the number of keystrokes needed to type the reference translation. Hence, the KSR is inversely related to the productivity increase which the system brings for the user.

```
Input: N_best_lists, Ref_Sentences, KSR=0, MAR=0
Output: KSMR
1: main()
2: {
3:    for (i=0; i< N_best_lists.size(); i++)
4:         Simulated_User (N_best_lists[i][0],i)
5:    KSMR=(KSR+MAR)/total_character*100
6: }

7: Simulated_User (char* Trans_offer ,int Id)
8: {
9:     Prefix=Find_biggest_prefix(Trans_offer
                                    , Ref_Sentences[Id])
10:    // Find_biggest_prefix compare two char*
11:    // and return the biggest identical substring
12:    if (Prefix== Ref_Sentences[Id])
13:    {
14:       KSR=KSR+1 // for accepting offer
15:       return ;
16:    }
17:    else
18:    {
19:       MAR=MAR+1 // for determining prefix by mouse
20:       Prefix= Prefix +Ref_Sentences[Id][ Prefix.size()]
21:       // the first non_match character is added to prefix.
22:       KSR=KSR+1 // for insert a character
23:       Simulated_User (Match_Prefix (Prefix,Id),Id)
24:    }
25: }

26: char* Match_Prefix(char* Prefix, int Id)
27: {
28:    min=1000
29:    index_min=-1
30:    for (i=0; i< N_best_lists[Id].size(); i++)
31:    {
32:       dis=Minimum_Edit_Distance(N_best_lists[Id][i]
                                    , Prefix)
33:    // Minimum_Edit_Distance is calculated by Levenshtein  Algorithm.
34:        if (dis<min )
35:       {
36:           min=dis
37:           index_min=i
38:       }
39:    }
40:    Suffix= N_best_lists[Id][ index_min] – Prefix
41:    return Suffix
42: }
```

Figure 2: The pseudo-code of the algorithm which simulates a human and matches prefix in the *N*-best list.

A KSR of 1 means that the interactive machine translation has never suggested an appropriate completion to the use sentence prefix, while a KSR value close to 0 means that the system has often suggested perfect completions.

MAR (Mouse-action ratio):

It is similar to KSR, but it measures the number of mouse pointer movements plus one more count per sentence (the user action needed to accept the final translation), divided by the total number of reference characters.

KSMR (Key-stroke and mouse-action ratio):

It is the summation of KSR and MAR, which is the amount of all required actions either by keyboard or by mouse to generate the reference translations using the interactive machine translation system divided by the total number of reference characters.

4.3. Experiments

In order to rescore the *N*-best list generated by the automatic speech recognizer, we make use of the translation models described in Section 3. The rescored *N*-best lists are used in the CAT system as N-best hypotheses lists. After human translator interact with the CAT and a prefix is formed, the CAT will search N-best hypotheses for founding a hypothesis which has minimum edit distance to the prefix and exactly includes the last (partial) word of the prefix. Then the CAT system returns remaining of target sentence to the user (from after last word to end of hypothesis). To study the effect of the N-best list size on the CAT results, we repeat the experiments with N-best lists which have a maximum of 1, 5, 10, 100, 1000 and 5000 hypotheses per sentence for the EPPS task. The results of the speech-enabled interactive-predictive CAT system are listed in Table 3 and 4.

Table 3: KSMR result for Test and Dev in percent. For each translation model, translation probability is calculated in one direction.

			Test	Dev
ASR		n=1	9.2330	12.4844
		n=5	7.8893	10.3986
		n=10	7.3995	9.7566
		n=100	6.3681	8.4446
		n=1000	5.7882	7.9736
		n=5000	5.6361	7.8683
SAR+MT	IBM1	n=1	8.5129	11.751
		n=5	7.1701	9.7380
		n=10	6.7058	9.1292
		n=100	5.7490	7.9496
		n=1000	5.3794	7.5926
		n=5000	5.2884	7.5205
	HMM	n=1	8.9872	12.247
		n=5	7.6180	10.152
		n=10	7.1501	9.5327
		n=100	6.0740	8.2896
		n=1000	5.5724	7.8164
		n=5000	5.4413	7.7057
	IBM3	n=1	8.4091	11.651
		n=5	7.1583	9.6807
		n=10	6.7623	9.0812
		n=100	5.7781	7.9456
		n=1000	5.3858	7.5879
		n=5000	5.3139	7.4903
	IBM4	n=1	8.1488	11.285
		n=5	6.9270	9.3283
		n=10	6.4764	8.7420
		n=100	5.5269	7.7808
		n=1000	5.2319	7.4292
		n=5000	5.1646	7.3556
	IBM5	n=1	7.9867	11.152
		n=5	6.7522	9.2268
		n=10	6.3872	8.7063
		n=100	5.4313	7.6987
		n=1000	5.2082	7.3951
		n=5000	5.1254	7.3308

Table 4: KSMR result for Test and Dev in percent. For each translation model, translation probability is calculated in two directions.

			Test	Dev
SAR+MT	IBM1 & IBM1-I	n=1	7.3686	9.8767
		n=5	6.4200	8.5220
		n=10	6.1487	8.0550
		n=100	5.4286	7.3339
		n=1000	5.1828	7.1325
		n=5000	5.1008	7.0582
	HMM & HMM-I	n=1	7.9385	11.014
		n=5	6.7395	9.2593
		n=10	6.4436	8.6382
		n=100	5.5842	7.6971
		n=1000	5.2702	7.4253
		n=5000	5.2046	7.3564
	IBM3 & IBM3-I	n=1	8.3099	11.248
		n=5	7.1146	9.4592
		n=10	6.6922	8.8450
		n=100	5.7472	7.8304
		n=1000	5.3566	7.4965
		n=5000	5.2884	7.4090
	IBM4 & IBM4-I	n=1	6.6749	9.2780
		n=5	5.8646	8.0410
		n=10	5.6088	7.6445
		n=100	5.0471	7.0489
		n=1000	4.8832	6.8506
		n=5000	4.8450	6.8212
	IBM5 & IBM5-I	n=1	6.7504	9.3662
		n=5	5.8974	8.1115
		n=10	5.6443	7.7606
		n=100	5.0872	7.1194
		n=1000	4.8960	6.8955
		n=5000	4.8678	6.8793

In spite of Table3 that shows the translation probability in one direction ($p(e_1^I|f_1^J)$). Additionally, in Table 4, for each translation model, we calculate the translation probability in both directions: $p(e_1^I|f_1^J)$ and$p(f_1^J|e_1^I)$. Both tables are shown the KSMR measure of the CAT.

4.4. Discussion

As the results show, there is a clear and significant accuracy improvement in all cases when moving from single-best to N-best translations. The best results obtained on the test and development sets are 5.13 % and 7.33 %, respectively. Both of results are produced by the IBM translation Model 5 and the N-best lists with maximum size 5000 hypotheses. According to these results, user of our CAT would only need an effort equivalent to typing about 5.13% and 7.33% of the characters in order to produce the correct translations for the test and development sets, respectively. These results are very ideal for CAT systems.

Also we could improve these results by using the translation models in both directions. These results are shown in Table 4. In this case, the best results obtained on the test and development sets are 4.87% and 6.88%, respectively. For better and easier comparing of the results, consider Figure 3 to Figure 6.

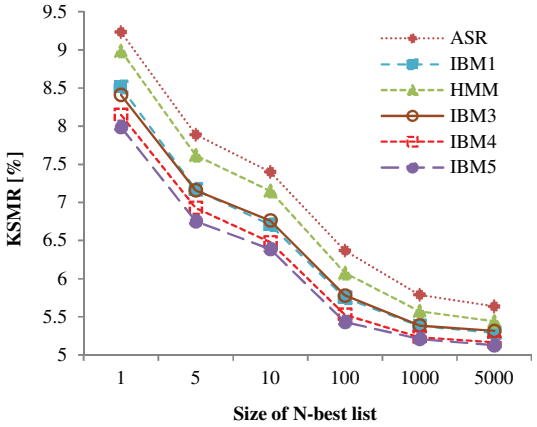

Figure 3: Results of the Interactive-predictive Speech-enabled CAT on the EPPS Test set.

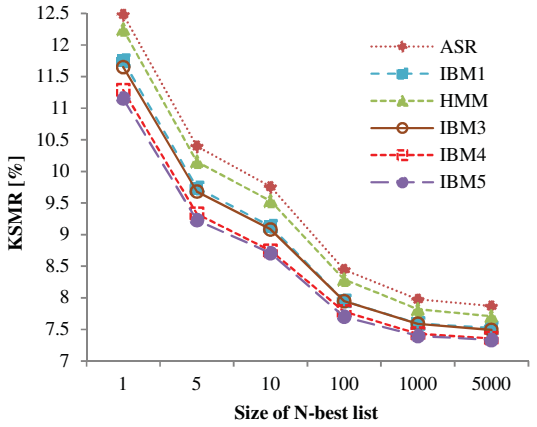

Figure 4: Results of the Interactive-predictive Speech-enabled CAT on the EPPS Dev set.

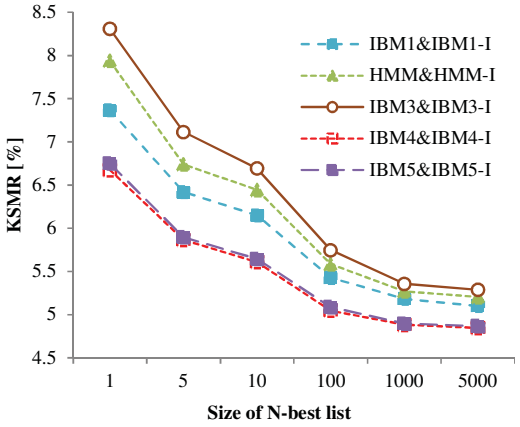

Figure 5: Results of the Interactive-predictive Speech-enabled CAT on the EPPS Test set.

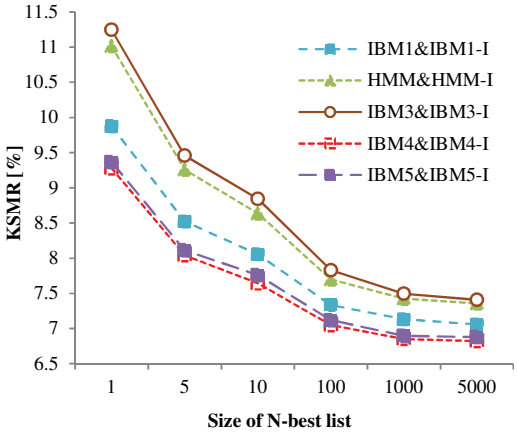

Figure 6: Results of the Interactive-predictive Speech-enabled CAT on the EPPS Dev set.

The successes obtained in these experiments are due to the quality of translations produced by the integrated ASR and MT systems and size of the N-best lists. With larger n-best list, the probability that the CAT system can suggest a better extension will increase.

5. Conclusion

The goal of this paper was to evaluate whether the accuracy of a speech-enabled interactive-predictive CAT system could be improved by using the N-best lists which are obtained by ASR and are rescored by translation models.

We introduced a general framework for integrating the speech recognition and translation models for automatic text dictation in the context of computer-assisted translation. We used the N-best lists which were produced by integrated ASR and MT systems, as N-best hypotheses in the CAT system and we achieved significantly better results.

References

[1]. Kay, M. (1973). *The MIND system,* in Natural Language Processing, pp. 155-188.

[2]. Brown, R.D., Nirenburg, S. (1990). *Human-computer interaction for semantic disambiguation,* In Processing of the International Conference on Computational Linguistics (COLING), PP. 42-47.

[3]. Maruyama, H., Watanabe, H. (1990). *An interactive Japanese parser for machine translation,* In Processing of the International Conference on Computational Linguistics (COLING), pp. 257-262.

[4]. Whitelock, P. J., McGee Wood, M., Chandler, B. J., Holden, N. and Horsfall, H. J. (1986). *Strategies for interactive machine translation: the experience and implications of the UMIST Japanese project*, In Proceedings of the International Conference on Computational Linguistics (COLING), pages 329-334.

[5]. Foster, G., Isabelle, P. and Plamondon, P. (1997). *Target-Text Mediated Interactive Machine translation*, in Kluwer Academic Publishers, pp. 175–194.

[6]. Langlais, P., Foster, G., and Lapalme, G. (2000). *TransType: a computer-aided translation typing system*, In Proceedings of the NAACL/ANLP Workshop on Embedded Machine Translation Systems, pp. 46–52.

[7]. Langlais, P., Lapalme G. and Loranger, M. (2002). *TRANSTYPE: Development–Evaluation Cycles to Boost Translator's Productivity*, in Kluwer Academic Publishers, pp. 77–98.

[8]. Foster, G. (2002). *Text Prediction for Translators,* Ph.D. thesis, Universit´e de Montr´eal, Canada.

[9]. Cubel, E., González, J., Lagarda, A. L., Casacuberta, F., Juan, A. and Vidal, E. (2004). *Adapting finite-state translation to the TransType2 project,* Proceedings of the Joint Conference combining the 8th International Workshop of the European Association for Machine Translation.

[10]. P. F. Brown, S. F. Chen, S. A. D. Pietra, V. D. Pietra, A. S.Kehler, and R. L. Mercer, "Automatic speech recognition in machine-aided translation", *Computer Speech and Language*, vol. 8, no. 3, pp. 177–187,1994.

[11]. M. Dymetman, J. Brousseau, G. Foster, P. Isabelle, Y. Normandin, and P. Plamondon, "Towards an automatic dictation system for translators: the TransTalk project", *in Proceedings of ICSLP-94*, pp. 193–196, 1994.

[12]. J. Brousseau, C. Drouin, G. Foster, P. Isabelle, R. Kuhn, Y. Normandin, and P. Plamondon, "French speech recognition in an automatic dictation system for translators: the transtalk project", *in Proceedings of Eurospeech*, pp. 193–196, 1995.

[13]. S. Khadivi, R. Zens and H. Ney, "Integration of Speech to Computer-Assisted Translation Using Finite-State Automata", *In Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 467–474, 2006.

[14]. S. Khadivi and H. Ney. 2, "Integration of Speech Recognition and Machine Translation", *in IEEE Transactions On Audio, Speech, And Language Processing*, VOL. 16, pp. 1551-1564, 2008.

[15]. Reddy, R. Rose and A. Désilets, "Integration of ASR and Machine Translation Models in a Document Translation Task", *In IEEE Transactions on Audio*, Speech, and Language Processing, Canada, 2007.

[16]. M. Paulik and A. Waibel, "Extracting clues from human interpreter speech for spoken language translation", *in Proc. ICASSP*, pp. 5097-5100, 2008.

[17]. E.Vidal, F. Casacuberta, L. Rodríguez, J. Civera, and C. Martínez. Computer-assisted translation using speech recognition. *IEEE Transaction on Audio, Speech and Language Processing*, 14(3):941-951, 2006.

[18]. K. A. Papineni, S. Roukos, and R. T. Ward, "Feature based language understanding, in EUROSPEECH", Rhodes, Greece, September, pp. 1435–1438, 1997.

[19]. K. A. Papineni, S. Roukos, and R. T. Ward, "Maximum likelihood and discriminative training of direct translation models", *in Proc. IEEE Int. Conf. on Acoustics*, Speech, and Signal Processing (ICASSP), vol. 1, Seattle, WA, pp. 189–192, 1998.

[20]. P. Beyerlein, "Discriminative model combination, in Proc. IEEE Int. Conf. on Acoustics", *Speech, and Signal Processing (ICASSP)*, vol. 1, Seattle, WA, pp.481 – 484, 1998.

[21]. F. J. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation", *in Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, PA, pp. 295–302, 2002.

[22]. F. J. Och, "Minimum error rate training in statistical machine translation", *in Proc. of the 41th Annual Meeting*

of the Association for Computational Linguistics (ACL), Sapporo, Japan, pp. 160–167, 2003.

[23]. P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L.Mercer, "The mathematics of statistical machine translation: Parameter estimation", *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[24]. F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models", *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[25]. F. J. Och, C. Tillmann, and H. Ney, "Improved alignment models for statistical machine translation*", in Proc. Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, University of Maryland, College Park, MD, pp. 20-28, 1999.

[26]. F. J. Och and H. Ney, "The alignment template approach to statistical machine translation", *Computational Linguistics*, vol. 30, no. 4, pp. 417–449, 2004.

[27]. Vogel, H. Ney, and C. Tillmann. HMM-based word alignment in statistical translation. In *Proceedings of the 16th conference on Computational linguistics*, pages 836–841, Morristown, NJ, USA, 1996.