Enhancing ASR by MT using Semantic information from Hindi WordNet

Aniruddha Tammewar, Karan Singla
LTRC, IIIT Hyderabad
India
{uttam.tammewar, karan.singla}@students.iiit.ac.in

Srinivas Bangalore
ATT Labs-Research
USA
srini@research.att.com

Michael Carl
CBS
Denmark
mc.ibc@cbs.dk

Abstract

In a conventional CAT (Computer Assisted Translation) system a human translator post-edits an automatically generated target language text using the keyboard. In this paper we extend a CAT system with speech input by which the translator speaks the translation, a process refered to as *sight translation*. We report several experiments to improve the performance of an automatic speech recognition system, taking advantage of machine translaton output and information from WordNet. Overall we outperform a baseline system which has no semantic information by an increased 1.6% word accuracy for the English to Hindi translation.

1 Introduction

To achieve high quality translations with better productivity and efficiency, post-editing of MT output and interactive computer assisted translation has been proposed [Kay, 1997; Langlais, 2002]. In an alternative setting, it has been shown that humans can translate nearly four times as quickly with little loss in accuracy simply by dictating, as opposed to typing, their translations [P.F Brown et al,1994]. In this paper, we describe a synthesis of both paradigms[3] in which we extend an interactive CAT system (CASMACAT [Ortiz-Martnez et al, 2012]) with speech as an additional input modality.

In SEECAT, the translation process is supported by an Automatic Speech Recognition (ASR) and a Machine Translation (MT) system, which transcribe the spoken speech signal into the target text and which assist the translator with partial translation proposals, predictions and completions on the computer monitor. An eye-tracking device follows the translators gaze path on the screen, detects where he or she faces translation problems and triggers reactive assistance.

This paper describes experiments to re-score ASR hypotheses for sight translation by using MT output and incorporating semantic information from Wordnet.

2 Previous Work

The idea of incorporating ASR and MT models was independently initiated by two groups of researchers at IBM [Brown et al., 1994], and in the TransTalk project [Dymetman et al., 1994; Brousseau et al., 1995]. The idea was to integrate the MT model with the ASR system, by combining the n-gram language model with the lexical translation probability of each word in the target language string.

More recently, work in ASR and MT integration was done by [Khadivi et al., 2005] and [Paulik et al..2005a ; Paulik et al..2005b], who tried different approaches for integrating MT models to ASR word graphs. They also studied the effect of late integration of ASR and MT models, along with various methods in which MT can be used as a language model for rescoring ASR.

In particular, [Khadivi et al., 2006] explored the benefit of integrating ASR and MT in a speech enabled CAT system using Finite State Transducer (FST) to search the composition. As the search in an ASR or a MT system is very complex, he showed that integrating using FST can decrease the word error rate without much increase in the complexity which is a must for real time scenario.

[3] A prototype of the SEECAT tool (Speech and Eye-gaze Enabled Computer Assisted Translation) was implemented during a summer project 2013, see url: http://bridge.cbs.dk/platform/?q=SEECAT

All of the above methods are based on an N-best rescoring approach.

In this paper, we study a method for integrating ASR and MT hypothesis which can be called *late-integration* as the integration is achieved after the hypotheses are generated. Differences in the vocabulary of the human translator (who speaks the translation) and the MT system may lead to icompatibility between the word forms produced by the ASR and MT systems, although on a conceptual level they may have the same meaning. To overcome such mismatches, we explore methods for using semantic information provided by Hindi WordNet (HWN) to rescore the ASR in a better way using MT.

The next few sections discuss a baseline system, methods of ASR and MT integration, experiments performed and the results achieved, along with discussing about the benefits of exploiting semantic information provided by Hindi WordNet.

3 Baseline Components

3.1 Automatic Speech Recognition

The Hindi ASR was trained on more than 20 hours of audio data (7k training sentences) with transcriptions. The data was collected from various sources as described in the Table 1.

Contributed By	Domain
KIIT	General text messages
McGill University	News
IIIT Hyderabad	Wikipedia Articles
SEECAT Workshop	Text messages, Tourism

Table 1: Hindi ASR Training Data

For a test set of 67 sentences from general domain, the word accuracy of ASR was 69.0%

3.2 Machine Translation System

The Statistical Machine Translation model for English-Hindi is trained using Moses [Koehn, et al,2007]; the training data used was collected from various sources as described in Table 2.

Contributors	Domain
ILCI [Choudhary et al]	Health and Tourism
John Hopkins [Post et al]	Wikipedia Articles
Other from web	General

Table 2: MT Parallel Training Data

A total of 140k cleaned up sentence pairs was used for training the MT model. The Language Model was training on the entire Hindi Wikipedia. The system gave a BLEU score of 21.33 for a evaluation data set of 500 sentences.

4 ASR and MT Composition

To integrate the ASR and MT output a common representation scheme is required. The ASR system takes an audio file as input and produces multiple hypotheses in the form of text, whereas the MT system takes a string in source language as input and produces multiple strings in the output. We represent both strings as word lattices i.e. *finite state transducers* (FST), similar to Mehryar Mohri et al, [2008], who used *weighted finite-state transducers* (FST) to represent different components of speech recognition system. The main reason for using FSMs is that the composition of two FSMs using edit machine is easier than other approaches. Both, ASR and MT produce multiple sequence of words along with probabilities which are encoded as weighted FST. For the operations on FSMs, we used the OpenFst[5] toolkit [Allauzen, Cyril, et al,2007].

4.1 Encoding ASR output as a FST

The ASR system generates multiple hypothesis strings for a given audio input, from which a FST is constructed.

Each hypothesis has a linear path in the FST, and each edge in the FST represents a word (input symbol) and its id in (output symbol). The last edge of each path has also an associated cost that ranks all the hypotheses. Figure 1 represents an example of the FST created from the 4 best hypotheses of a sentence. After creating the FST, it is determinised and minimised.

To make things clear, let's take an example which will be followed in the other sections also.

Example

Source English String : *Children were playing in the garden*

Ref: *b*ache pArka me khela rahe the

The ASR hypotheses :

1. मर्चेंट चैंबर मे खेल रहे थे
   merchant chamber me khela rahe the
2. बच्छेद्रीपाल में खेल रहे थे
   bchaadripal me rahe the
3. बच्चे पर हक मे खेल रहे थे
   bache par me khela rahe the
4. बच्चे पार्क मे खेल रहे थे
   bache pArka me khela rahe the

[5] http://www.openfst.org

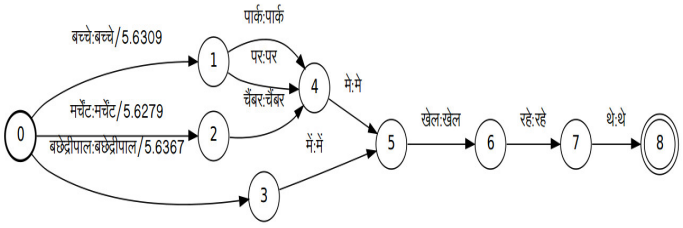

Figure 1: Sample ASR FST

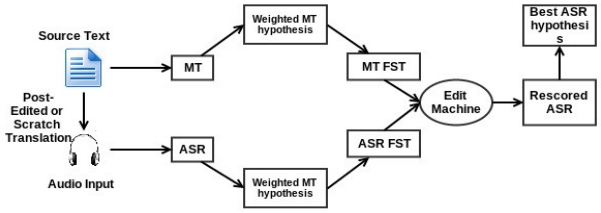

Figure 3: ASR and MT Composition

4.2 Encoding MT output as a FST

In a similar manner described in the previous subsection, the MT system also produces multiple hypothesis for a given input string. Figure 2 describes the translated English string to Hindi in the form of a FST representing 4 hypotheses.

Example

Source English String : *Children were playing in the garden*

The MT hypotheses :

1. बच्चों की इस बाग में खेल रहे थे
   bacchon kI es bAga mein khela rahe the
2. बच्चों की इस बाग में खेल रहे थे
   bacchon kI es bAga mein khela rahe the
3. इस बाग में खेल रहे बच्चों की
   es bAga mein khela rahe bacchon kI
4. इस बाग में खेल रहे थे , चिल्ड्रेन
   es bAga mein khela rahe the , children

4.3 Composition

An edit machine composes the ASR and MT lattices into one FTM and successively computes the edit distance between the MT hypothesis and the ASR hypothesis. The ASR hypotheses are rescored during composition, according to the edit distances calculated from the MT hypotheses. Smaller edit distance represents more similarity between ASR and MT hypotheses and hence better score for that ASR hypothesis. More common words between the ASR and the MT hypotheses results in lesser edit distance. In this way, the ASR output is biased towards the words from MT hypotheses. Figure 3, describes how the composition of is carried out. Table 3 shows an example where the composition gives a better reranking of the ASR hypotheses.

In Table 3, the 1-best in third column (composition of ASR and MT) has a better score than ASR alone as it will have better word accuracy, when compared to reference in the first column and less edit distance with MT hypothesis, as compared to the highest ranked in ASR alone. The details about the last column will be discussed in section 5.

Reference	ASR Hypothesis	ASR+MT Hypothesis	(ASR+MT) Synset Hypothesis
बच्चे पार्क मे खेल रहे थे	मर्चेंट चैंबर मे खेल रहे थे	बच्चे पर हक मे खेल रहे थे	बच्चे पार्क मे खेल रहे थे
	बछेद्रीपाल में खेल रहे थे	बछेद्रीपाल में खेल रहे थे	बच्चे पर हक मे खेल रहे थे
	बच्चे पर हक मे खेल रहे थे	मर्चेंट चैंबर मे खेल रहे थे	बछेद्रीपाल में खेल रहे थे
	बच्चे पार्क मे खेल रहे थे	बच्चे पार्क मे खेल रहे थे	मर्चेंट चैंबर मे खेल रहे थे

Table 3: Sample Hypotheses

In all the experiments, the variation of N followed similar pattern. All the systems performed best at the value of N to be 10. At any value of greater than 10, the performance was poor.

5 Using Wordnet for Composition of ASR and MT hypothesis

As the data on which the ASR and the MT systems are trained are from different domains, the words in the MT hypotheses space differ from those produced by the ASR system. As a result, there may be multiple alternative translations for each word of the English source sentence produced by the MT system, but the translator may speak out a translation which is not contained in the MT output. In such cases the mismatch between different word-forms affects the edit-distance calculations, and results in a sub-optimal rescoring of the ASR hypotheses. However, if the system gets information about synonym relations, it can match different words although their orthography is different and hence score hypotheses higher when containing the same concepts. The required semantic information can be extracted from the Hindi WordNet.

5.1 Hindi Wordnet

Hindi WordNet (HWN) is a lexical database developed under the IndoWordNet project [Debasri, Chakrabarti, et al] in line with the English Wordnet. For each lexical entry in the WordNet, multiple senses corresponding to each of the four POS categories (Noun, Adjective, Adverb, Verb) are

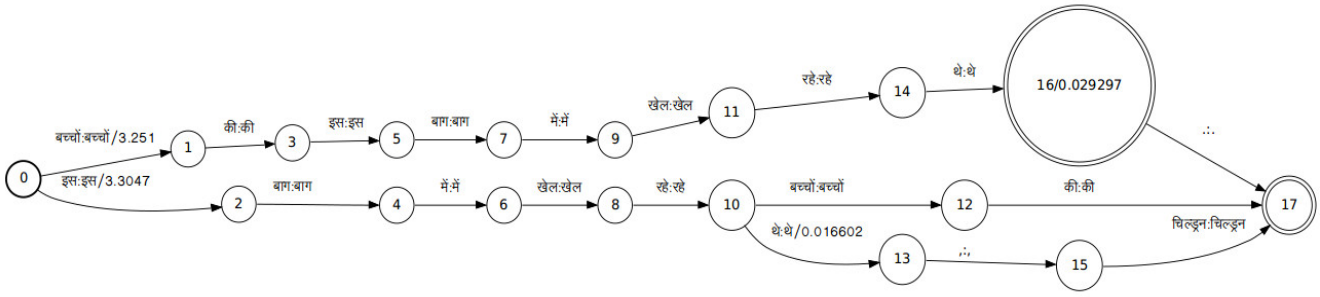

Figure 2: Sample MT FST

present. A synset is a group of synonyms i.e words with the same meaning. Each synset has a unique synset ID assigned to it. If we replace the words in the ASR and MT lattices with their corresponding synset IDs, the words which are synonyms of each other will be represented by the same synset ID. When we make the composition with such lattices, the ASR hypotheses containing the words which are synonyms of the words in the MT hypotheses will be ranked higher, as the orthography of the synonyms is now same, and results in a better rescoring of ASR hypotheses. If we take the highest scoring hypothesis after this composition, the string will contain the synset IDs instead of words.

We performed different experiments based on the sense selection from a WordNet entry, the retrieval of lexical items from the synset IDs and different strategies of evaluating the output. These strategies are explained in the following subsection.

5.2 Sense Selection

Attributed to the phenomenon of lexical ambiguity, a lexical item can have senses varying across contexts. HWN lists all the possible senses for all the possible syntactic categories of a lexical item. To choose the contextually appropriate sense of a lexical item is a challenging task. In the following, we discuss our approach to select the sense of a lexical item best suited in a given context.

5.2.1 Category Based Sense Selection

Consider a word *'chaat'*, it can either mean *'lick'* (Verb) or *'snacks'* (Noun). The syntactic category of a lexical item provides an initial cue for the sense selection from the varied senses of a lexical item. Among the varied senses, we filter out the senses that do not fall into the syntactic category denoted by the POS-tag of the lexical item.

5.2.2 Intra-Category Sense Selection

Words are ambiguous not only across different syntactic categories but also within a same category. For example, the word खाना (khaana) has six different senses in the Noun category and 12 different senses in the Verb category in the HWN. The Noun can mean *'meal'* or *'box- a rectangular drawing* or *'drawer'* or one of the other three Noun senses. If a lexical item has many subcategories, they are investigated with different strategies.

5.2.2.1 First Sense: Among the varied senses, we select the first sense listed in HWN corresponding to the POS-tag of a given lexical item. The choice is motivated by our observation that the senses of a lexical item are ordered in the descending order of their frequencies of usage. The first sense listed in HWN is thus the predominant sense of the given lexical item. Let us consider an English source sentence '*I made the meal*', let us say that MT gives the translation as 'मैंने भोजन बना लिया' (Maine bhojana banaa liyaa) and one of the ASR hypothesis is 'मैंने खाना बना लिया' (Maine khaana banaa liyaa), here the human translator and MT are using different words for the same concept of *'meal'*. If we replace the words in these strings with the synset IDs corresponding to the first sense, the two words 'भोजन' (bhojana) and 'खाना' (khaana) would fall in the same synset ID 1830, therefore at the time of rescoring, the ASR hypothesis containing 'खाना' will be ranked higher.

5.2.2.2 Merge Senses: In this strategy we merge all the senses listed in HWN corresponding to the POS-tag of the given lexical item. The motivation behind this strategy is that the senses in the HWN for a particular word-POS pair are too finely grained, so a given pair of words which denotes the same concept in the given context,

may be distributed over different synsets. Also, if we choose the first sense for replacement, the synset ID of the first sense of a word from a pair of synonyms may be different from the synset ID of first sense of the other word. For example let us consider the two words 'बाग' (baaga) and 'पार्क' (paarka). The synset corresponding to the first sense (synset ID: 3534) of the word 'पार्क' contains the words: bagiicaa, bagQiicaa, baaga, vaatikaa, bagiyaa, udyaana, baarii, upavana, apavana, baagQ, **paarka**, baadii. But if we do the same thing for 'पार्क', the first synset(synset ID: 27458) contains the words: पार्क. The sense corresponding to the first sense of 'बाग' is ranked 7th in the sense-list of 'पार्क'. Therefore the replacement with first synset would not result in the same orthography for the two synonyms.

We overcome this problem by merging all the senses corresponding to both the words i.e. form a new artificial synset which contains all the lexical items present in all the synsets listed in the HWN entry for 'बाग' and assign it some synset ID (say 3534), maybe the ID of the first synset. We use this synset ID if any of the word from this group occurs in the remaining string. This approach will be successful even if the senses in the HWN entry for a given word are totally different from each other like in the case of word 'चाट' (chaat) having two different meanings 'lick' or 'snack'. Though it can have two meanings, but in a given context it will have only one meaning either 'lick' or 'snack'. The context for MT and ASR would always be the same for the same sentence in the consideration, therefore the words used by ASR and MT hypotheses should be represented by the same synset ID.

5.3 Retrieval of Lexical Item from Synset IDs

After we compose the lattices with synset IDs and rescore the ASR hypotheses, we can select the best scoring hypotheses, but the hypothesis string will contain synset IDs instead of the lexical items. Now, we have to retrieve the corresponding lexical items from these synset IDs. We have multiple lexical items for a given synset ID, which also includes the words produced by ASR and MT. We choose the word produced by ASR, because this word would be orthographically closer to what the speaker has said, than the word produced by MT. For example, there are many word forms in the Synset ID 3534 including 'बाग' and 'पार्क'. We chose to retrieve 'पार्क' as it was generated by the ASR system.

5.4 Example

We will consider the same example we have taken for the ASR and MT composition. We will choose the merging strategy for sense selection.

The new ASR hypotheses with synset IDs :

1. merchant chamber syn_27146 syn_4716 rahe the
2. syn_7083 syn_752 syn_2732 syn_27146 syn_4716 rahe the
3. syn_7083 syn_752 mein syn_672 rahe the
4. syn_7083 syn_3534 syn_27146 syn_4716 rahe the

The new MT hypotheses with synset IDs:

1. syn_7083 syn_27092 es syn_3534 syn_23048 syn_4716 rahe the .
2. syn_7083 syn_27092 es syn_3534 syn_23048 syn_4716 rahe the
3. es syn_3534 syn_23048 syn_672 rahe syn_7083 syn_27092
4. es syn_3534 syn_23048 syn_4716 rahe the, children

Best Hypothesis after composition :
syn_7083 syn_752 mein syn_672 rahe the
Best hypothesis after retrieving words :
बच्चे पार्क मे खेल रहे थे
bache pArka me khela rahe the

6 Evaluation Strategies and Results

In this section, we explore different strategies for the evaluation of our approach and present the results of the evaluation.

6.1 Hypothesis and Reference string lexical matching

Once we get the final hypothesis string, with the words retrieved from Synset IDs, we compare it with the reference string at lexical level, and produce the Word Error rate and Word Accuracy. For calculating WER and Word Accuracies we are using SCLite toolkit which comes with CMU Modelling ToolKit [Rosenfeld et al].

6.2 Replace words in the reference string with the synset IDs

In this method, instead of retrieving the lexical items from the final hypothesis containing synset IDs, we replace the words in the reference string with the corresponding synset IDs using corresponding strategy for sense selection. The results calculated with this strategy can considered to be a oracle score i.e. the best possible results that we can reach using the synset technique.

Experiment	Hypothesis String	Reference String	Word Accuracy (sequence)	Word Accuracy (unweighted bag of words)
Baseline ASR	Words from ASR	Words	69.0%	69.0%
Baseline ASR+MT	Words from ASR	Words	70.8%	71.7%
Synset ASR+MT(first sense)	Words from ASR	Words	71.3%	72.0%
Synset ASR+MT(merged synsets)	Words from ASR	Words	72.4%	72.2%
Oracle (Best Possible)	Synset IDs	Synset IDs	74.5%	73.3%

Table 4: Scores

6.3 Results and Discussion

For the experiments we selected a test set that consists of 67 sentences recorded by 3 native speakers of Hindi. Error rate was computed using the standard evaluation tool SCLite. Table 4 represents word accuracy score for ASR, ASR and MT baseline integration, and the experiments in which information from Wordnet was used. We performed two types of expeiments: (1) we are taking both ASR and MT hypotheses as sequence of words and forming FSTs as described in section 4.1, and (2) we are considering ASR hypotheses as sequences and MT hypotheses as unweighted bags of words. With this, there are two baseline systems for 'ASR and MT integration', one with MT as a sequence of words and another with unweighted bag of words. Overall, experiments showed an increase of word accuracy of 3.4% over the baseline ASR output. By looking at the results we cannot comment which of the two strategies is better: MT as a sequence or as a bag of words. The results also show that the 'merge senses' technique performs better over 'first sense' technique.

The oracle score word accuracy came out to be 74.5% and 73.3% for the 'sequence of words' and 'unweighted bag of words' respectively. With the help of the synsets information, we have reached close to the oracle (best possible using sysets) word accuracy.

In Hindi, a word can often have different orthographic forms, as e.g. 'चाँद'(chaand) and 'चांद', which have similar pronunciation and are valid forms. This creates problems if the orthography of the word is different in hypothesis and the reference string.This problem is common with 'chandra-bindu', 'anuswar' and 'n'(न्). In the evaluation, this word is considered as wrong word which affects the word accuracy.

7 Conclusion

In this paper we showed that in a CAT system with speech as input method, the performance of ASR can be improved using semantic information from wordnet. This technique can also be applied for all the language pairs for which there is a similar knowledge base available for target langauge.

For many language pairs like Hindi-English not much parallel data is available, which produces a problem of data sparsity. In MT applications and CAT this problem can be handled partially with replacing source and target words by synset ID's. In our case, we need not worry about retrieving words from ID's as we will take words from ASR hypotheses. This will improve the efficiency of machine translation and the overall system performance.

References

Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. 2007. Openfst: A general and efficient weighted finite-state transducer library. In *Implementation and Application of Automata*, pages 11–23. Springer.

Sergio Barrachina, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, et al. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.

Julie Brousseau, Caroline Drouin, George F Foster, Pierre Isabelle, Roland Kuhn, Yves Normandin, and Pierre Plamondon. 1995. French speech recognition in an automatic dictation system for translators: the transtalk project. In *Eurospeech*.

Peter F Brown, Stanley F Chen, Stephen A Della Pietra, Vincent J Della Pietra, Andrew S Kehler, and

Robert L Mercer. 1994. Automatic speech recognition in machine-aided translation. *Computer Speech & Language*, 8(3):177–187.

Narayan Choudhary and Girish Nath Jha. TODO. Creating multilingual parallel corpora in indian languages.

Chakrabarti Debasri, Narayan Dipak Kumar, Pandey Prabhakar, and Bhattacharyya Pushpak. 2002. Experiences in building the indo wordnet: A wordnet for hindi. In *Proceedings of the First Global WordNet Conference*.

Martin Kay. 1997. Machine translation: The disappointing past and present. In *Survey of the state of the art in human language technology*, pages 248–250. Cambridge University Press.

Shahram Khadivi, András Zolnay, and Hermann Ney. 2005. Automatic text dictation in computer-assisted translation. In *INTERSPEECH*, pages 2265–2268.

Shahram Khadivi, Richard Zens, and Hermann Ney. 2006. Integration of speech to computer-assisted translation using finite-state automata. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 467–474. Association for Computational Linguistics.

Philipp Koehn, Marcello Federico, Wade Shen, Nicola Bertoldi, Ondrej Bojar, Chris Callison-Burch, Brooke Cowan, Chris Dyer, Hieu Hoang, Richard Zens, et al. 2006. Open source toolkit for statistical machine translation: Factored translation models and confusion network decoding. In *Final Report of the 2006 JHU Summer Workshop*.

Philippe Langlais and Guy Lapalme. 2002. Trans type: Development-evaluation cycles to boost translator's productivity. *Machine Translation*, 17(2):77–98.

Mehryar Mohri, Fernando Pereira, and Michael Riley. 2002. Weighted finite-state transducers in speech recognition. *Computer Speech & Language*, 16(1):69–88.

Daniel Ortiz-Martınez, Germán Sanchıs, Francisco Casacuberta, Vicent Alabau, Enrique Vidal, José-Miguel Benedı, Jesús González-Rubio, Alberto Sanchıs, and Jorge González. 2012. The casmacat project: The next generation translators workbench. In *Proceedings of the 7th Jornadas en Tecnologıa del Habla and the 3rd Iberian SLTech Workshop (IberSPEECH)*, pages 326–334.

Matthias Paulik, S Stuker, C Fugen, Tanja Schultz, Thomas Schaaf, and Alex Waibel. 2005a. Speech translation enhanced automatic speech recognition. In *Automatic Speech Recognition and Understanding, 2005 IEEE Workshop on*, pages 121–126. IEEE.

Matthias Paulik, Christian Fügen, Sebastian Stüker, Tanja Schultz, Thomas Schaaf, and Alex Waibel. 2005b. Document driven machine translation enhanced asr. In *INTERSPEECH*, pages 2261–2264. Citeseer.

Matt Post, Chris Callison-Burch, and Miles Osborne. 2012. Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 401–409. Association for Computational Linguistics.

Ronald Rosenfeld and Philip Clarkson. 1997. Cmu-cambridge statistical language modeling toolkit v2.