Tense and Mood Decision with Similarity Search in Japanese to Spanish Machine Translation

Manuel Medina González and Tsutomu Endo*

Abstract—**Tense and mood are pieces of information normally unattended in Japanese to Spanish machine translation as they do not exist formally in the former language. In this paper we propose a new technique to solve this issue by using a tree distance function and the nearest neighbor approach in order to find similar sentences in a knowledge base. A query sentence is input and it is transformed into a tree using a language model; then, a series of similar sentences is retrieved from the knowledge base; the most similar is selected and all the information of the predicates in it is assigned to the predicates in the query sentence. Our technique proves to be accurate as it obtained a 61% of correct results, just below the 63% obtained by Systran.**

Keywords: ***machine-translation, Japanese, Spanish, similarity-search, aspect***

1 Introduction

In machine translation, one of the most difficult issues is to generate enough information to create elements that do not exist in the source language but they do in the target language. It is clear that traditional approaches, like figure 1 where an *interlingua language* is used, are not enough to solve the issue, because no matter how deep the analysis might be performed, those elements will never appear in the interlingua, leading to loss of information at the moment of the generation stage.

When translating from Japanese to Spanish, *tense* and *mood* are important pieces of information for the message to be communicated properly. However, the different nature of both languages makes difficult to integrate this information at the moment of the translation, as there are 3 tenses in Japanese (present, past and future) while in Spanish there are 16 [3, 4]; moreover, the concept of mood does not exist in Japanese. Thus, it becomes necessary the creation of techniques that allow that information to appear at the moment of the analysis. These techniques must be thought from the point of view of the target language (Spanish in our case). The process of that type of techniques is shown in figure 1.

*Kyushu Institute of Technology, Kawazu 680-4 Iizuka, 820-8502, Japan. Email: manuel@dumbo.ai.kyutech.ac.jp, endo@pluto.ai.kyutech.ac.jp

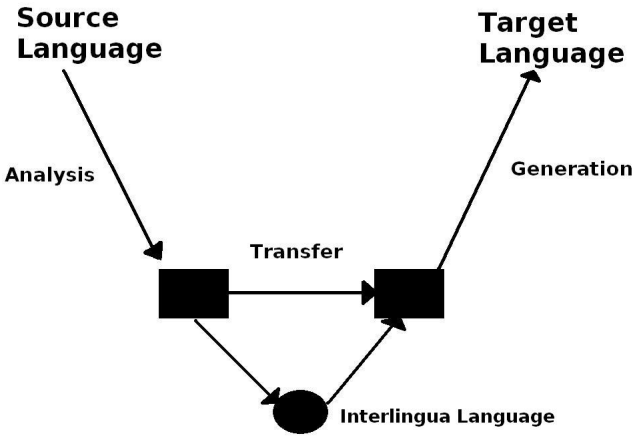


Figure 1: Traditional machine translation approach

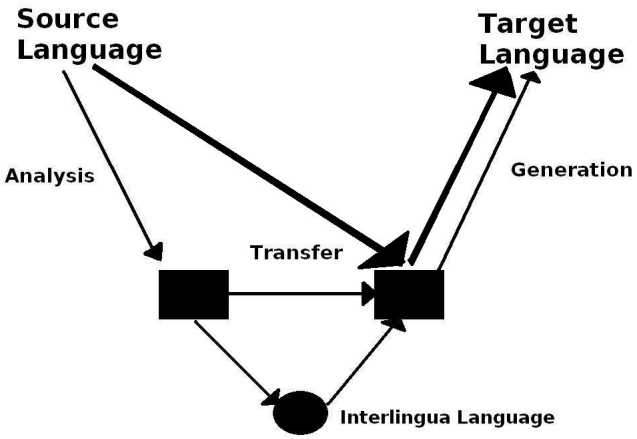


Figure 2: Approach considering features in the target language

In this paper we propose such a technique. Our method uses *similarity search* and the *nearest neighbor* approach to search for sentences similar to the input, retrieve the information about their predicates and assign it to the predicates of the input.

The rest of this paper is organized as follows: in section 2 we present previous works on tree distance functions and on Japanese to Spanish machine translation analysis. In section 3 we show the language model we use for this paper and explain how we represent the data and the features we consider. In section 4 we present our technique and the implemented algorithm. In section 5 we show the results of the experiments we performed to test our technique, and we also discuss about them and what our future work is. Last, in section 6 we present our conclusions on the subject.

2 Background

There is a large number of researches regarding tree distance functions. Most of them focus on the computation complexity and the time it takes to finish a calculation. The first algorithm for calculating tree edit distance (*TED*) was presented by Tai [16]. Shasha and Zhang proposed an algorithm to reduce the complexity of the original algorithm to $O(n^4)$ [15], and the algorithm presented by Demaine *et al.* [2] achieved a worst-case $O(n^3)$ time algorithm when two trees have size n.

On the other hand, the number of previous works regarding the study of tense and mood in Japanese to Spanish machine translation is still very reduced. We presented an naive approach to solve *subjunctive mood* in [9]. However, that technique is based on rules, thus, new rules must be created whenever a new case is discovered. The study of subjunctive mood from the point of view of linguistics is not new. May we suggest to consult [18, 5, 1] for further information.

To our knowledge, there are no previous works that use similarity search to specifically deal with tense and mood decision in Japanese to Spanish machine translation.

3 Data Representation

3.1 Language model

The text to be analyzed is represented as a tree structure based on the language model presented in [10] and depicted in figure 3.

We performed a modification in the *sentence* category: originally, a sentence is a series of *bunsetsu*[1] related to the same predicate; in this paper, we added an *auxiliary verb* to the sentence. By using an extra verb, we can separate the modifiers of the verb and the auxiliary verb

[1]the smallest meaningful structure in Japanese

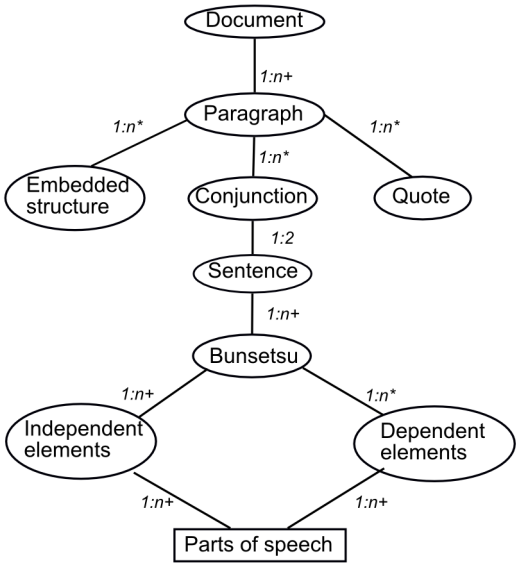


Figure 3: The language model. Japanese parts of speech are not shown.

and treat them separately. Figure 4 depicts a case where an auxiliary verb appears.

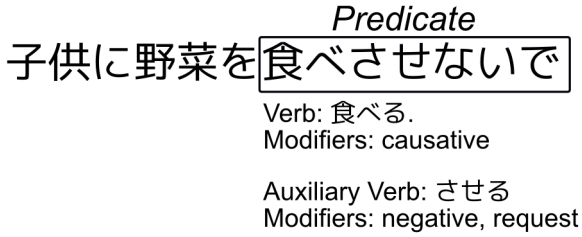


Figure 4: Example of a sentence with auxiliary verb. In this case, the Japanese auxiliary verb “させる” generates the auxiliary verb

.

It is important to note that not all the Japanese auxiliary verbs generate auxiliary verbs in Spanish.

3.2 Predicate modifiers

For each predicate in the structure, we attach a list of modifiers to it. They are based mainly on the dependent elements [17] contained in the predicates. At the moment, the list includes 40 modifiers, but this number is not definitive and can be increased (or decreased) as necessary. Due to the space limitation, we will not list here all of them. Table 1 shows some modifiers and the condition that must be met for them to be set.

3.3 String representation

We use *treelib* [13] implementation of the tree distance functions to perform the calculations. This library needs

Table 1: An extract of the predicate modifiers considered for this paper

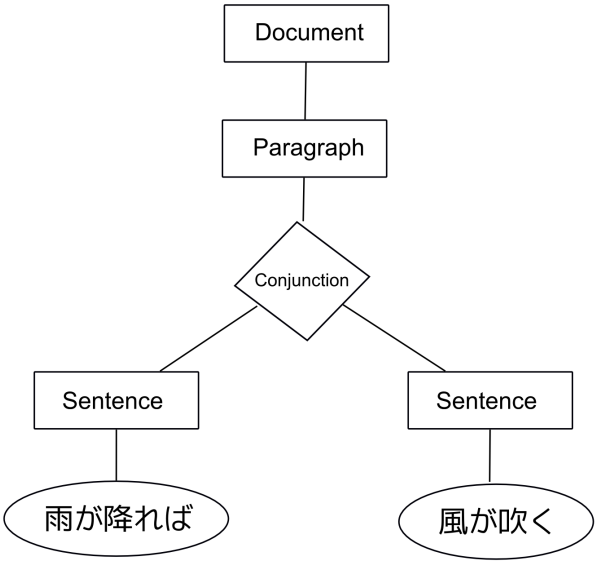
Modifier	Condition to be set
formal	auxiliary verb ます (masu) is present
negative	auxiliary verb ぬ (nu) is present (as ぬ (nu, ない (nai), ず (zu) or ん (n))
passive	auxiliary verbs れる (reru) or られる (rareru) are present
causative	auxiliary verbs せる (seru) or させる (saseru) are present
volitive	auxiliary verb たい (tai) is present
past	Predicate ending in た (ta)
tara	Predicate ending in たら (tara)

the trees are represented as strings of the form *a(b,c)*, where *a* is the root node and *b* and *c* are its daughter nodes.

At the moment of the conversion, we consider the following points:

- The root node, a document, is not included. It is always the same. Although it does not affect the result, it takes more time to be calculated.
- Only the nodes that are significant for tense and mood resolution are converted. This includes: paragraphs, quotes, conjunctions, embedded structures and predicates. Adverbs of frequency or time, like いつも (always), and nouns that express time, like 明日 (tomorrow), are also converted.
- Predicate modifiers are converted as children of the predicate nodes.

Figure 5 shows the representation of the sentence "雨が降れば、風が吹く" (if it rains, the wind blows) in the language model and in the string representation. The categories below "sentence" were omitted.



Paragraph(Sentence(Verb(BA)),
Sentence(Verb(PLAIN_FORM)))

Figure 5: The representation of the sentence in the language model and as string

4 Similarity Search

4.1 Tree distance function

In this research, we used the tree distance function *MTD* proposed by Müller-Molina *et al.* [14] to perform the distance calculation. MTD has been used successfully in semantic program matching [12]. In contrast to traditional TED algorithm, MTD gives more importance to changes near the leaves. This behavior is desirable because we use predicate modifiers and insert them as leaves, and these changes are more significant than the occured in nodes near the root.

4.2 The technique

Our technique can be summarized in the following steps:

1. Convert the queried sentence into a rooted tree.
2. Transform that tree into its string representation.
3. Calculate the distance from the queried sentence to each element in the *knowledge base* using MTD.
4. Check the obtained distances and select the nearest element.
5. Assign the information of the predicates of the selected element to the predicates of the queried sentence.

Figure 6 depicts the flow of the technique, and the process is shown in algorithm 1.

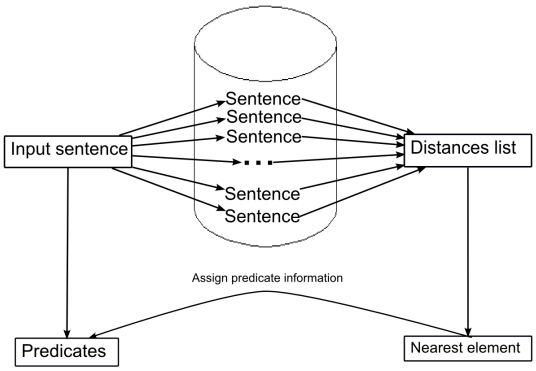


Figure 6: An overall view of the technique

In the case there is more than one element with the shortest distance, we implemented a voting system between all the nearest elements. The voting process is as follows:

Algorithm 1: Nearest neighbor algorithm for tense and mood decision

```
1  procedure Decide Tense and Mood(Sentence query)
2      i : Integer, SList : Array of String
3      Distances : Array of Double
4      String query_sentence_tree
5      SList ← Retrieve data from knowledge base
6      query_sentence_tree ← Convert to
       String-Tree(query);
7      i ← 0
8      while i < |SList| do
9          Distances[i] ← MTD(new_sentence_tree, s_i)
10     end
11     sort(Distances)
12     nearest ← Get first element from Distances
13     Each predicate in new ← Retrieve predicates
       information(nearest)
14 end procedure
```

1. Get all the elements with the shortest distance.
2. Search for the most common predicate information among them.
3. Assign the information of the found predicates to the queried sentence.

4.3 Example

Let us suppose that we want to find the correct tense and mood for the predicates in the sentence 誰にとっても世界が平和であるほうがよい (it is better for everyone to have peace). After representing it in the language model, we get a structure like figure 7.

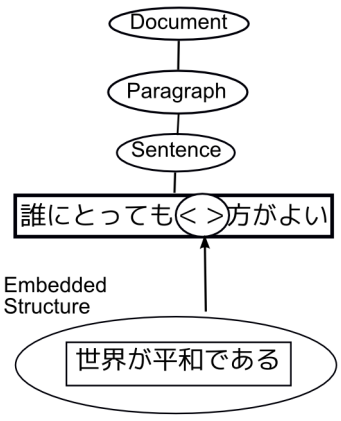


Figure 7: The sentence represented in the language model. Categories were simplified

After converting it to a string, and once we have calculated the distance against all the sentences in the knowledge base, we get the sentence with the shortest distance value. In this case: 私は住民との合意なしに市がゴミ処理施設を許可するのには反対です (I'm against the goverment allowing the construction of a garbage disposal facility without the agreement of the citizens). As we can see in figure 8, the structure is basically the same.

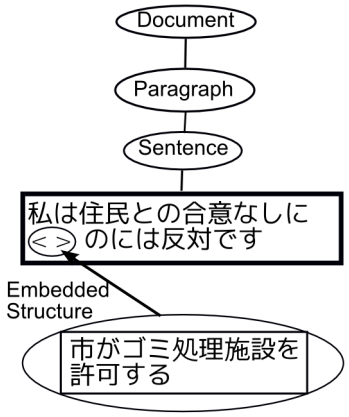


Figure 8: Structure of the sentence selected from the knowledge base as the nearest element.

We retrieve the stored information of the predicates in this sentence and assign it, in order, to the original sentence. In this case, we find out that the first predicate should be in *present tense, subjunctive mood*, and the second in *present tense, indicative mood*. As this is the expected result, the predicates of the input sentence have now the correct information regarding tense and mood.

5 Experiments and Results

In order to test how the method behaves, we implemented a prototype system and prepared a knowledge base with 300 sentences extracted from [7]. The data behavior was tested using the *10-fold cross validation* method [6]. A total of 707 predicates were analized. We also performed the distance calculation with ted, using the implementation presented by Shasha and Zhang [15]. To compare the results, we translated the sentences using *Systran* and analyzed the tense and mood given to the predicates.

The results obtained using nearest neighbor are shown in table 2. Table 3 shows the results obtained using the voting system. Finally, table 4 shows the percentage obtained after analyzing the results obtained with Systran.

Table 2: Results with nearest neighbor.

Function	Min	Max	Mean	Std. Deviation
MTD	51%	69%	61%	0.053773
TED	52%	68%	56%	0.054270

Table 3: Results using voting system.

Function	Min	Max	Mean	Std. Deviation
MTD	45%	64%	54%	0.063581
TED	47%	66%	57%	0.056384

Table 4: Percentage of correct results with Systran.

Program	Correct Results
Systran	63%

5.1 Discussion

The use of similarity search provides a simple but powerful way to decide the tense and mood of the predicates in a sentence. The results obtained show the accuracy of the method for the provided dataset. However, it is important to notice that the obtained results do not reflect the accuracy of a final translation because we are not generating sentences. We focus only on tense and mood. It is necessary to analyze more elements if we want to output a translation.

Providing enough information for each of the predicates included in a sentence leads to an improvement in the accuracy of the obtained translation. Our research stands on *analysis* and *transfer* stages only. With better and more detailed information in these steps, the generation stage may output a more accurate translation.

To our knowledge, the existent machine translation systems do not output specific information about the features we based our research on; however, the results obtained by Systran help us to understand better the value of the results obtained with similarity search. These results can still be improved. There were some cases where, although the trees of both elements, query and training, were the same (the distance between them was 0), the results were not completely correct because of the difference of the predicates expected. Figure 9 depicts an example of such a case.

As we created the trees with minimal data, one possible solution is to apply rules when creating the trees that add more information to the node about verbs or patterns that may use subjunctive mood. We can give more weight to those nodes in the tree distance function in order to make the distance between similar structures increases, allowing to other similar structures to be analyzed and considered.

If a paragraph contains more than one sentence, or a sentence contains an embedded structure or a quote, we also consider to analyze them separately by applying the same method recursively. Parallel processing is possible, but, in some cases, the real tense and mood of a predicate

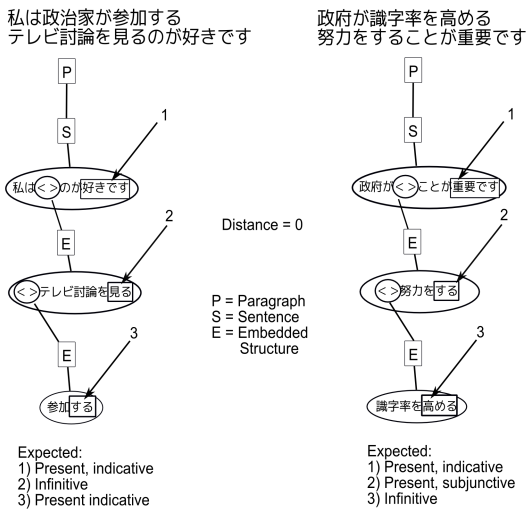


Figure 9: Example of sentences with the same structure but different predicates expected

in a sentence are decided by the combination and the role they play in that sentence. In such cases, we could combine the technique proposed in [8] with the presented in this paper to get better results. This is part of our future work.

6 Conclusions

In this paper, we presented a new technique to solve the tense and mood in Japanese to Spanish machine translation. Our method uses similarity search with tree distance functions to find the distance between a sentence and all the sentences included in a knowledge base. Then, it uses the nearest neighbor technique to get the most similar. Once a sentence has been selected, the information of its predicates is assigned to the predicates of the new sentence. The method proved to be satisfactory as it reported a mean of 61% of correct results, just below the 63% obtained by Systran. Similarity search provides a powerful, yet simple way to work with grammatical issues in sentences, and we plan to continue using it to improve the results presented in this paper.

References

[1] Aoife Ahern and Manuel Leonetti. The spanish subjunctive: Procedural semantics and pragmatic inference. In Rosina Márquez-Reiter and María Elena Placencia, editors, *Current Trends in the Pragmatics of Spanish*, pages 35–57. John Benjamins Publishing Company, 2004.

[2] Erik D. Demaine, Shay Mozes, Benjamin Rossman, and Oren Weimann. An optimal decomposition algorithm for tree edit distance. In *In Proceedings of the 34th International Colloquium on Automata, Lan-*

guages and Programming (ICALP, pages 146–157, 2007.

[3] Juan Luis Fuentes de la Corte. *Gramática moderna de la lengua española*. Limusa, 2007.

[4] Álex Grijelmo. *La gramática descomplicada*. Taurus, 2008.

[5] Mark Jary. Mood in relevance theory: a re-analysis focusing on the spanish subjunctive. *UCL Working Papers in Linguistics*, 14:157–187, 2002.

[6] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. pages 1137–1143. Morgan Kaufmann, 1995.

[7] Kazuyoshi Koike. スペイン語作文の方法 *(How to Write Compositions in Spanish)*. 第三書房, 2002.

[8] Manuel Medina González. Spanish Case Information Analysis and its Application to Japanese to Spanish Machine Translation. Master's thesis, Kyushu Institute of Technology, 2006.

[9] Manuel Medina González and Hirosato Nomura. An Approach to Spanish Subjunctive Mood in Japanese to Spanish Machine Translation. In *AI2007: Advances in Artificial Intelligence*, pages 744–748, 2007.

[10] Manuel Medina González and Hirosato Nomura. A Japanese Language Model with Quote Detection by Using Surface Information. In *MICAI 2008: Special Session*, pages 65–71, 2008.

[11] Maite Melero. Combining machine learning and rule-based approaches in spanish and japanese sentence realization. In *Second International Natural Language Generation Conference*, 2002.

[12] Arnoldo José Müller Molina and Takashi Shinohara. On approximate matching of programs for protecting libre software. In ACM Press, editor, *CASCON '06*, pages 275–289, 2006.

[13] Arnoldo José Müller-Molina. Treelib, library of tree distance functions. http://treelib.berlios.de/index.html.

[14] Arnoldo José Müller-Molina, Kouichi Hirata, and Takeshi Shinohara. A tree distance function based on multi-sets. In *ALSIP'08, PAKDD Workshops (To appear in LNCS)*, pages 90–100, 2008.

[15] Dennis Shasha and Kaizhong Zhang. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal of Computing*, 18:1245–1262, 1989.

[16] Kuo-Chung Tai. The tree-to-tree correction problem. 26(3):422–433, 1979.

[17] Junichi Tajika. くわしい国文法 *(Detailed Japanese Grammar)*. 文英堂, 東京, 2002.

[18] H. Ueda. 日本語の『は』とスペイン語の接続法 (the japanese particle 'wa' and the spanish subjunctive mood). 日本語学, 21(7), 2003.