Dependency-based Analysis for Tagalog Sentences *

Erlyn Manguilimotan and Yuji Matsumoto

Graduate School of Information Science
Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara, Japan, 630-0192
{erlyn-m, matsu}@is.naist.jp

Abstract. Interest in dependency parsing increased because of its efficiency to represent languages with flexible word order. Many research have applied dependency-based syntactic analysis to different languages and results vary depending on the nature of the language. Languages with more flexible word order structure tend to have lower performances compared to more fixed word order languages. This work presents, for the first time, a dependency-based parsing for Tagalog language, a free word order language. The parser is tested using manually assigned POS and auto-assigned POS data. Experiment results show an average of about 78% accuracy on unlabeled attachment scores, and about 24% on complete dependency scores. As the position of the sentence head is not fixed, we look into sentence head accuracy as assigned by the parser. Results show that around 83% - 84% of the sentence head are assigned correctly.

Keywords: Dependency parsing, Tagalog, free word order, morphological information

1 Introduction

Morphologically rich languages (MRL) are languages where word-level information affects syntactic relations (Tsarfaty et al., 2010). Recently, MRL have drawn attention in statistical parsing. While state-of-the-art parsers brought high performance scores for languages such as English and Japanese, this is not true for MRLs. Dependency parsing techniques and approaches are developed for languages with relatively fix word orders; however, they are not successful when applied for MRL. In the 2007 CoNLL Shared Task on Dependency parsing, languages such English and Catalan have top scores while languages with free word order and higher degree of inflection have lower scores (e.g. Arabic, Basque)(Nivre et al., 2007). Although factors such as small data can contribute to the degradation of the parser performance, linguistic factors such as morphological inflection and freeness in word order have made it difficult to parse MRL with models and techniques developed with languages like English (Tsarfaty et al., 2010).

In this paper we look into a state-of-art dependency parser applied on Tagalog language. Prior to this work, there is no known data-driven dependency-based parsing for Tagalog. Tagalog is the national language of the Philippines and is a member of the Austronesian language family. It is a morphologically rich language which has affixations, stress shifting, consonant alteration, and reduplication (Nelson, 2004). Affixations include prefixation, infixation, suffixation and circumfixation, where two or three affixes are attached to a root word. Highly inflected verbs show verbal focus *(actor, object, etc.)* and aspect *(perfective, imperfective, and contemplated)*.

Aside from its morphological complexity, sentences are word order free. In Tagalog sentences, predicate comes first for pragmatically unmarked clauses but the NP complements take flexible positions in the sentence (Kroeger, 1993). Figure 1 showing possible orders of the complements

---

25th Pacific Asia Conference on Language, Information and Computation, pages 343–352
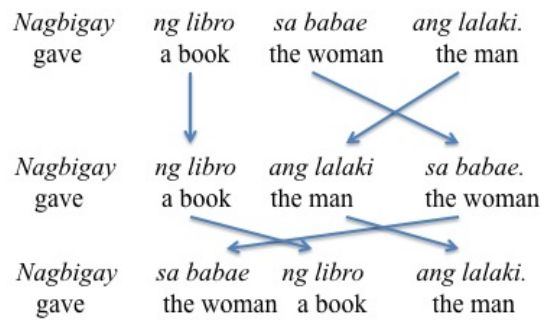
Figure 1: Three (3)of the possible translations of the sentence *"The man gave the woman a book"*

in the Tagalog sentence *"Nagbigay ng libro sa babae ang lalaki"* (The man gave the woman a book). According to Schachter and Otanes (1972) as mentioned in Kroeger (1993), *"the sentences include exactly the same components, are equally grammatical, and are identical in meaning"*. On the other hand, as briefly explained in Section 3, adverbial and nominal phrases may take pre-verbal position.

For Tagalog sentences, dependency-based syntactic parsing is preferred over phrase-structure based analysis because dependency analysis does not rely on word positions(Tsarfaty et al., 2010), instead, sentence representations are based on links between words, called dependencies.

Due to limited resource, this work deals only with unlabeled data. Our corpus are annotated with dependency heads but do not include dependency relations of the word (e.g.NMOD, ROOT). We look into the *unlabeled attachment scores* (UAS), or the number of correct heads, as well as *complete scores*. We also look into the number of sentence heads correctly predicted. The free word order nature of Tagalog may put a verb at the initial position or after a nominal or adverbial element in the sentence.

We briefly introduce dependency parsing in section 2, section 3 briefly explains sentence structures in Tagalog and section 4 discusses the how heads are assigned to words in a sentence. Sections 5 describes the experiment set-up and results. Finally, we discuss some issues found in the parsing results in section 6.
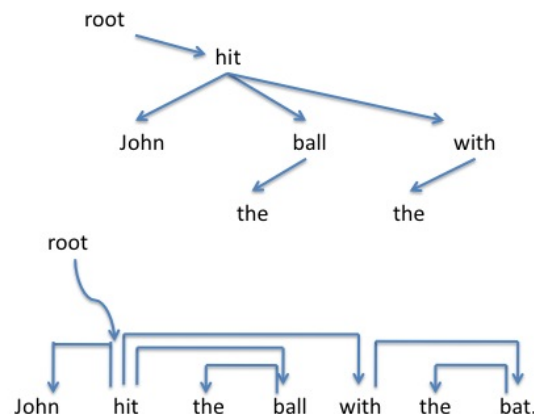

Figure 2: An example of a dependency structure. (Source: McDonald and Pereira, 2006)

2 Dependency Parsing

Dependency analysis creates a link from a word to its dependents. When two words are connected by a dependency relation, one takes the role of the *head* and the other is the *dependent* (Covington, 2001). The straightforwardness of dependency analysis has been used in other NLP tasks such as word alignment (Ma et al., 2008) and semantic role labeling (Hacioglu, 2004). Dependency parsers developed were either graph-based (McDonald and Pereira, 2006) or transition-based parsers (Nivre and Scholz, 2004; Yamada and Matsumoto, 2003). Syntactic analysis for different languages also use dependency parsers such as in Japanese (Kudo and Matsumoto, 2001; Iwatate et al., 2008), English (Nivre and Scholz, 2004), Chinese (Chen et al., 2009; Yu et al., 2008) to mention some.

The task of dependency-based parsing is to construct structure for an input sentence and identify the syntactic head of the word in the sentence(Nivre et al., 2007). Figure 2 shows a sentence with dependency arcs (below) and its equivalent dependency tree structure(top). Each word in the sentence has one syntactic head except for the root or the head of the sentence. Dependency trees can be projective or non-projective. It is projective if no arcs cross over when drawn above the sentence, while it is non-projective when there are edges crossing each other. Non-projective dependency is a result of parsing languages with more flexible word order such as German and Czech(McDonald and Pereira, 2005).

In this work, we applied the graph-based maximum spanning tree(MST) parser of McDonald and Pereira (2006) on Tagalog language. In MST, a sentence as $\boldsymbol{x} = x_1 \cdots x_n$ is represented as a set of vertices $V = v_1 \cdots v_n$ in a weighted directed graph G. A dependency tree y of a sentence is a set of edges (i, j), where $(i, j) \in E$ in the graph G, $E \subseteq [1 : n] \times [1 : n]$ and edge (i, j) exists if there is a dependency from word x_i to word x_j in the sentence. The *maximum spanning tree* of graph G is a rooted tree $y \subseteq E$ that maximizes the score $\sum_{(i,j) \in y} s(i, j)$, where $s(i, j)$ is the score of a dependency tree y. For detailed explanation of MST, please refer to McDonald and Pereira (2006).

3 Tagalog Sentences

3.1 Pre-verbal Position

As shown in Fig. 1, Tagalog NP complements take flexible position in the sentence. However, nominal and adverbial elements may take pre-verbal position (Kroeger, 1993) such as the sentences in Fig. 3.

Itong	*tasa*	*ay*	***binili***	*ko*	*sa*	*pamilihan.*
This=LNK	*cup*	*INV*	*PERF=buy*	*SG.GEN.*	*DAT=market*	

I bough this cup at the market.

(a) *ay*-inversion

Itong	*tasa,*	***binili***	*ko*	*sa*	*pamilihan.*
This=LNK	*cup*	*PERF=buy*	*SG.GEN.*	*DAT=market*	

This cup, I bought at the market.

(b) Topicalization

Figure 3: Sentences with pre-verbal elements (Source: Kroeger, 1993, p. 123)

3.2 Non-verbal Predicates

In addition to verbs, prepositional, nominal and adjectival phrases take predicate positions in the sentence (Kroeger, 1993) . Figure 4 shows two sentences with non-verbal predicates. Sentence

Bago　　　*ang mga kotse=ng*　　　*iyon.*
New　　　NOM=PL=car=LNK　　　*that.*
Those cars are new.

(a) Predicate Adjective

Opisyal　　　*sa=hukbo*　　　*ang panganay.*
Officer　　　DAT=army　　　NOM=eldest.
The eldest child is an officer in the army.

(b) Predicate NP

Figure 4: Examples of sentences with non-verbal predicate (Source: Kroeger, 1993, p. 131)

4a has an adjective *"bago"* (new) as predicate while sentence 4b has the NP *"opisyal sa hukbo"* (officer in the army) as predicate.

Other linguistic explanations of Tagalog structures can be found in (Kroeger, 1993). Nagaya (2007) also explains Tagalog syntax and pragmatics and how they interact in Tagalog clauses.

4　Dependency Heads

From the part-of-speech annotated corpora available, we manually assigned heads to words in the sentences. Deciding *"What heads what?"* may sound simple, however, often it is not. Nivre (2005) discussed some criteria in distinguishing the head and dependent relations. For Tagalog, we applied a *head-complement*, *head-modifier*, *head-specifier* relations principle.

In general, predicate comes before its complements, thus, annotating such sentences is straightforward, such as in Fig. 5 for the sentence *"Bumili ang lalaki ng isda sa tindahan"* (The man bought a fish at the store). The verb *bumili* (bought, actor focus) heads the complements *lalaki*(man), *isda* (fish) and *tindahan* (store) and *ang*, *ng* and *sa* are dependents of the noun they specify. Head-specifier relation refer to the determiner and noun relation.
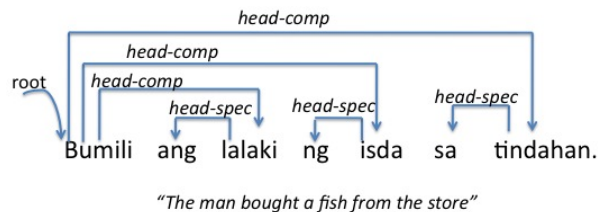

Figure 5: Dependency relations between words in the sentence *"Bumili ang lalaki ng isda sa tindahan"*

4.1　Modifiers

Modifiers may appear in a linear order relative to the modified phrase (noun phrase or verb)(Kroeger, 1993). They are connected with the noun phrase using a linker "na" such as in "mabait *na* bata" (good kid). However, when a modifier ends with a vowel or the letter "N" such as "matalino" (smart) and "antokin" (sleepy), the modifier is suffixed with *-ng* or *-g* (i.e. *"matali-nong bata"*(smart kid), *antoking bata*(sleepy child)). The latter case has a more direct annotation, while the former has to be decided which heads linker *"na"*. In this case, since the existence of linker "na" depends on the modifier, we assigned the modifier as the head of the linker. Figures 6a and 6b illustrates these dependencies.
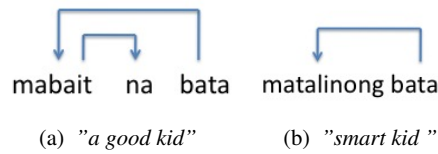
346

(a) *"a good kid"*   (b) *"smart kid "*

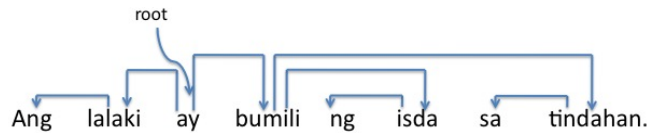Figure 6: Dependencies for modifiers and nouns

Figure 7: Dependency structure for the sentence with "ay"- inversion

4.2 Sentence Head

In case of a verbal predicate, the head of the VP phrase is the head of the sentence. However, Section 3 explains different positions of verb in a sentence. One pre-verbal position of a sentence is the use of lexeme *ay*. The sentence in Fig. 5 can be written as *"Ang lalaki **ay bumili** ng isda sa tindahan"* as in Fig. 7. The meaning of the sentence did not change; however, the nominative argument *ang lalaki* is placed before the verb *bumili*(bought). According to Kroeger (1993), *ay* could be treated as a kind of sentence-level auxiliary. In this case, if a sentence is in an *ay*-inversion format, *ay* is assigned as the *head* of the sentence instead of the verb. The nominative argument marked by determiner *ang* becomes a sibling of the verb. Sentences with non-verbal predicates follow the basic head-modifier rule in case it is adjectival or PP predicates, making the head of the modified phrase the head of the sentence.

5 Experiment

5.1 Data

The sentences used in the experiment were extracted from POS annotated Tagalog corpora consisting of novels and short articles. We only chose sentences of lengths L, for $4 \geq L \leq 20$. Longer sentences have tendencies of lower accuracies while shorter sentences tend to perform better (McDonald and Nivre, 2008). A total of 2,741 sentences were used for the experiments. The total number of sentences were then divided into 5 parts for a 5-fold cross validation. Table 1 gives the data distribution in the 5-fold cross validation while Table 2 shows the distribution of the Test and Training data by sentence lengths. Compared to other treebanks, Tagalog data set is relatively small; however, this is the start in building Tagalog Dependency Treebank for future work.

Table 1: 5-fold Cross Validation Data Distribution

Data	Test	Train
1	548	2,192
2	548	2,192
3	548	2,192
4	546	2,194
5	550	2,190

Number of Sentences per Part

347

Table 2: Test Data Distribution by Length

Data/Lengths	Test Data			Train Data		
	4-10	11-15	16- 20	4-10	11-15	16- 20
1	295	148	105	1,192	569	432
2	297	145	106	1,190	572	430
3	260	159	129	1,227	558	407
4	302	141	103	1,186	576	433
5	333	124	93	1,155	593	443

Table 3: POS Tagging Results

Test Data	POS Accuracy
1	89.09
2	87.18
3	88.84
4	89.37
5	90.46
average	88.96

5.2 Experiment Set-up

5.2.1 POS Tagging. Auto-assigned part-of-speech (POS) tagging is done using a conditional random fields (CRF) based tool CRF++[1]. We used a hierarchical tag set for Tagalog with 88 specific tags derived from 10 word categories (verb, noun, adjective, etc.)[2]. The same data set described in Section 5.1 are used in training and testing the POS tagger. POS tagging result is given in Table 3.

5.2.2 Parsing. We performed dependency parsing using McDonald and Pereira (2006)'s MST Parser. To be able to run the parser with morphological features, we used the CoNLL-X shared task format[3]. The parser is trained with fine-grained and coarse-grained POS tags. Coarse-grained POS, fine-grained POS, word stem (lemma) and morphological features (affixes and reduplication) are used as input to the parser. In our experiments, we used two word related feature settings. First, we run the parser with word stem feature and no morphological features, which we refer as MST 1 in Table 4; then we run the parser using word stem and morphological features (affixes and reduplication) (MST 2 in Table 4). Summary of features for the parser is presented in Table 4. We also run our parsers with *first - order* and *second - order* dependencies. *First-order* dependency scores are limited to single edge dependency while *second-order* dependency takes the score of two adjacent edges. Please refer to McDonald and Pereira (2006) for further explanations on first-order and second-order dependencies.

5.3 Result

The Tagalog parsers are tested using manually assigned POS tags and automatically tagged POS data. The manually assigned data is considered as the gold data in this work. Dependency parsing results are shown in Table 5 for the manually assigned POS tags and Table 6 for the auto-tagged POS. The parser is evaluated using unlabeled attachment score (*UAS*), or the accuracy of the total

---

[1] CRF++ by Taku Kudo http://crfpp.sourceforge.net/

[2] Unpublished work

[3] CoNLL-X website http://ilk.uvt.nl/conll/

Table 4: Parser Features

CPOSTAG =coarse-grained POS tag, POSTAG=fine-grained POS tag

MST1	*word form (or punctuation), lemma (word stem), CPOSTAG, POSTAG*
MST2	*word form (or punctuation), lemma (word stem), CPOSTAG, POSTAG , morphological info (affixes, reduplication)*

Table 5: Test results using manually assigned POS tags

Test Data	MST 1				MST 2			
	1st Order		2nd Order		1st Order		2nd Order	
	UAS	Complete	UAS	Complete	UAS	Complete	UAS	Complete
1	0.787	0.253	0.792	0.246	0.788	0.239	0.790	0.239
2	0.785	0.251	0.788	0.261	0.772	0.228	0.775	0.242
3	0.781	0.206	0.783	0.204	0.771	0.201	0.777	0.202
4	0.779	0.222	0.783	0.250	0.779	0.248	0.777	0.231
5	0.755	0.230	0.761	0.246	0.759	0.241	0.755	0.235
Ave	0.777	0.232	**0.781***	**0.241**	0.773	0.231	0.774	0.229

* at $\alpha = 0.05$, MST1 2nd order UAS is significant

number of correct heads assigned to each word. *Complete* dependency analysis is also computed, or the number of sentences with (all) correct dependencies.

Affixes and reduplication information have helped in deciding the POS tags (Manguilimotan and Matsumoto, 2009), thus we expected the same for parsing. However, this is not the case. In Table 5, results show that 2nd-order MST1 features obtained the highest accuracy in both *UAS* and *complete* scores for manually-tagged data compared to the test results using MST2 features. It is to be noted that MST1 features do not include affix and reduplication information while MST2 has these features as described in Table 4. This may imply that correct POS information have helped choose the head of the word and the use of other morphological information have harmed the performance of the parser. On the other hand, the auto-tagged POS data in Table 6 shows different results. The 2nd-order MST2 has the highest *UAS* but 2nd-order MST1 and 1st-order MST2 has the highest in *complete scores*. The error in POS tagging (about 11%) have affected the parsing results, but the morphological information (MST2) may have helped in choosing the correct head of the word.

Aside from dependency accuracies, we also looked at the sentence head selection accuracy. Here, we only considered correct sentence heads as chosen by the MST parser. Results are presented in Tables 7 and 8. The parser is about 84% accurate in choosing the sentence head for test data with gold POS tags and about 83% for auto-tagged POS. Heads of shorter sentences were mostly chosen correctly by the parser compared to those of longer sentences. Also, sentences with heads in initial positions have mostly predicted correctly than those with pre-verbal elements such as adverbial phrases, except for sentences with lexeme *"ay"* as heads where they were mostly correctly predicted.

Table 6: Test results using auto-assigned part-of-speech tags

	MST 1				MST 2			
Test Data	1st Order		2nd Order		1st Order		2nd Order	
	UAS	Complete	UAS	Complete	UAS	Complete	UAS	Complete
1	0.767	0.213	0.770	0.218	0.767	0.218	0.778	0.211
2	0.756	0.228	0.760	0.233	0.758	0.226	0.756	0.220
3	0.761	0.178	0.761	0.187	0.753	0.187	0.761	0.184
4	0.748	0.217	0.749	0.213	0.759	0.222	0.760	0.215
5	0.733	0.202	0.737	0.219	0.741	0.221	0.740	0.219
Ave	0.753	0.207	0.755	**0.214**	0.756	**0.214**	**0.759***	0.209

at $\alpha = 0.05$, MST2 2nd order UAS is NOT significant

Table 7: Sentence Head Selection Accuracy with Manually Tagged POS

	MST 1		MST 2	
Test Data	1st Order	2nd Order	1st Order	2nd Order
1	85.40	86.50	85.95	86.86
2	82.66	82.30	81.39	80.84
3	84.12	84.12	83.21	83.39
4	81.45	81.27	80.73	80.73
5	88.12	87.93	88.48	88.85
Ave	84.35	84.43	83.95	84.13

6 Discussions

We looked at some words that we believed to have lower chances of ambiguity, for example, determiners (DT) *ang* (the) and *mga* (plural marker for noun) considering that these do not require morphological information. These two function words come together often to indicate plurality, such as in "**ang mga** bata" *(the children)* or "**ang mga** bahay" *(the houses)*. However, when there are modifiers between the determiner *ang* or *ang mga* and the noun, the parser assigned wrong head to the determiners. These cases may be isolated accounting part of the 10% error for determiners in our data; however, this is still observable considering that distances between head and dependent should not be a problem in MST parser.

Word categories that topped the errors in head selection are noun(NN), conjuncts(CC), adjective(JJ) and adverb(RB). For NN, proper nouns such as titles (e.g. "doktor tiburcio" *(doctor tiburcio)*, "kapitan basilio" *(captain basilio)*) or a noun modifying another noun (e.g. "balkonahe=ng kawayan" *(bamboo balcony or balcony made of bamboo)*) are some cases that are found to be inconsistent.

On the other hand, lexeme "ng" is seen as ambiguous since it can function as a genetive marker in "kumain **ng** pan ang bata" *(the child ate bread)*, where "ng" is dependent on "pan/NN" (bread), or as a possessive marker in "ang ina **ng** bata" *(the child's mother)*; in this case, "ng" is dependent on "ina/NN" (mother). In both cases "ng" is headed by a noun. The same is observed with "sa", which can be a dative marker as in sentences in Fig. 3 or as part of an adverbial head phrase such as in "para sa" *(for – benefactive)*, "dahil sa" *(because of – causative)*.

Although we did not include in Section 5.3, we initially run our parsers with sentence lengths of greater than 30 words. Scores of those experiments are lower than the reported results in this paper. This reiterates that shorter sentences perform better than longer sentences (McDonald and Nivre, 2008). MST parser should not have a problem with distance, however, ambiguities in sentences cannot be helped.

Table 8: Sentence Head Selection Accuracy with Auto-Assigned POS Tags

	MST 1		MST 2	
Test Data	1st Order	2nd Order	1st Order	2nd Order
1	86.31	84.85	86.49	85.95
2	78.83	79.56	80.65	79.92
3	85.40	84.12	83.94	83.94
4	80.65	81.38	81.56	81.75
5	86.91	85.27	88.00	88.72
Ave	83.62	83.03	84.12	84.05

7 Summary

This paper introduces the first dependency-based syntactic analysis of Tagalog, a morphologically rich and word order free language. We described the scheme we used in assigning the head of the word. A state-of-the art dependency parser, MST parser, is used in our experiment. Experiments are performed manually assigned POS, which is considered as the gold data, and automatically tagged POS. When gold POS information are used, the parser tends to perform better when there are no morphological information used as features. However, when auto-tagged POS are used, the parser with morphological information has higher accuracy. This may indicate that morphological information helps when POS tagging is not accurate.

Morphological information will be more interesting to examine in labeled dependency analysis, especially verb affixes, because this will determine what kind of relationship does the verb have with its arguments. However, our work only deals with unlabeled dependencies, thus semantic relations of words are not included in our work.

Although dependency parsing is a more fitting framework than phrase-structure analysis in dealing with languages with flexible word order it still does not perform better compared to languages with a more fixed word order. Parsing morphologically complex languages with free word order gives rise to issues on the right data annotation, or what kind of morphological features are to be included in the parser(Tsarfaty et al., 2010).

The parsing results for Tagalog is still not high enough compared to other languages. Whether the factors of dependency parsing for Tagalog is the small sized data or the inadequacy of features, we still have to know. As our future work, we aim to increase the size of the Tagalog dependency treebank we have started. And since our current work deals only with unlabeled data, the next step is to run parser with labeled dependency relations and see which morphological features best solve issues in Tagalog dependency parsing.

References

Bonus, D.E. 2003. *The Tagalog Stemming Algorithms (TagSA).* In the Proceedings of the Natural Language Processing Research Symposium, DLSU, Manila.

Chen, W., D. Kawahara, K. Uchimoto, Y.Zhang, and H. Isahara. 2009. *Using Short Dependency Relations from Auto-Parsed Data for Chinese Dependency Parsing.* ACM Transactions on Asian Language Information Processing (TALIP), volume 8 issue 3.

Covington, M. 2001. *A Fundamental Algorithm for Dependency Parsing.* In Proceedings of the 39th Annual ACM Southeast Conference, pp. 95-102.

Iwatate, M., M. Asahara, and Y. Matsumoto. 2008. *Japanese Dependency Parsing Using a Tournament Model.* In Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pp. 361-368.

Hacioglu, K. 2004. *Semantic Role Labeling Using Dependency Trees* In Proceedings of the 20th International Conference on Compurational Linguistics.

Kroeger, P. 1993. *Phrase Structure and Grammatical Relations in Tagalog.* Center for the Study of Language and Information, Stanford California.

Kudo, T. and Yuji Matsumoto. 2002. *Japanese Dependency Analysis using Cascaded Chunking* In Proceedings of the Conference on Computational Natural Language Learning (CoNLL-2002).

Ma, Y., S. Ozdowska, Y. Sun, and A.Way. 2008. *Improving Word Alignment Using Syntactic Dependencies.* In Proceedings of the Second ACL Workshop on Syntax and Structure in Statistical Translation (SSST-2), pp. 69-77.

Manguilimotan, E. and Matsumoto, Y. 2009. Factors Affecting POS Tagging for Tagalog. *In the Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation*, pp. 803-810 (2009).

McDonald, R. and F. Pereira. 2005. *Non-projective Dependency Parsing using Spanning Tree Algorithms.* In the Proceedings of Human Language Technology and Empirical Methods in Natural Language Processing(HLT-EMNLP), pp. 523-530.

McDonald, R. and F. Pereira. 2006. *Online Learning of Approximate Dependency Parsing Algorithms.* In the Proceedings of the European Chapter of the Association for Computational Linguistics (EACL).

McDonald, R. and J. Nivre. 2008. Integrating Graph-based and Transition-based Dependency Parsers. *Association for Computational Linguistics (ACL).*

Miguel, D. and Roxas, R.E.. 2007. *Comparative Evaluation of Tagalog Part-of-Speech Taggers.* In the Proceedings of the Proceedings of the 4th National Natural Language Processing.

Nagaya, N. 2007. *Information Structure and Constituent Order in Tagalog.* Language and Linguistics 8.1:343-372, 2007.

Nelson, H. 2004. *A Two-Level Engine for Tagalog Morphology and a Structured XML Output for PC-KIMMO.* Available : [Online: http://contentdm.lib.byu.edu/ETD/image/etd465.pdf] .

Nivre, J. and M. Scholz. 2004. *Deterministic Dependency Parsing of English Text* In Proceedings of the 20th International Conference on Compurational Linguistics.

Nivre, J. 2005. *Dependency Grammar and Dependency Parsing.* MSI report 05133 MSI report 05133. VŁxj University: School of Mathematics and Systems Engineering.

Nivre, J., J. Hall, S. Kubler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. *The CoNLL 2007 Shared Task on Dependency Parsing.* Proceedings of the CoNLL Shared Task Session on EMNLP-CoNLL 2007, pp 915-932.

Schachter, P. and F. Otanes. 1972. Tagalog Reference Grammar. *University of California Press.*

Tsarfaty, R., et al. 2010. Statistical Parsing of Morphologically Rich Languages (SPMRL) What, How and Whither. *In Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically Rich Languages*, pp. 1-12.

Yamada, H. and Y. Matsumoto. 2003. *Statistical Dependency Analysis with Support Vector Machines.* In Proceedings of 8th International Workshop on Parsing Technologies, pp. 195-206.

Yu, K., D. Kawahara, and S. Kurohashi. 2008. *Chinese Dependency Parsing with Large Scale Automatically Constructed Case Structures.* In the Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pp. 1049-1056.