[1]Syntactic tools for text watermarking

Hasan M. Meral[1], Emre Sevinç[3], Ersin Ünkar[2], Bülent Sankur[4], A. Sumru Özsoy[1,3], Tunga Güngör[2,3]
[1]Linguistics Program, Boğaziçi University,
[2] Department of Computer Engineering, Boğaziçi University,
[3] Cognitive Science Program, Boğaziçi University,
[4]Department of Electrical and Electronic Engineering, Boğaziçi University,
{mesut.meral, bulent.sankur, ozsoys, gungort}@boun.edu.tr}, emres@bilgi.edu.tr,
eiunkar@gmail.com,
Telephone: (90) 212 359 6414, Fax: (90) 212 287 2465, Bebek, İstanbul, Turkey
Corresponding author: Hasan Mesut Meral

ABSTRACT

This paper explores the morphosyntactic tools for text watermarking and develops a syntax-based natural language watermarking scheme. Turkish, an agglutinative language, provides a good ground for the syntax-based natural language watermarking with its relatively free word order possibilities and rich repertoire of morphosyntactic structures. The unmarked text is first transformed into a syntactic tree diagram in which the syntactic hierarchies and the functional dependencies are coded. The watermarking software then operates on the sentences in syntax tree format and executes binary changes under control of Wordnet to avoid semantic drops. The key-controlled randomization of morphosyntactic tool order and the insertion of void watermark provide a certain level of security. The embedding capacity is calculated statistically, and the imperceptibility is measured .using edit hit counts.

Key words: Natural language watermarking, text watermarking, morphosyntax

1. INTRODUCTION

Text watermarking is an emerging technique in the intersection of natural language processing and the technologies of forensics and security. Text watermarking aims at embedding additional information in the text itself with the goals of subliminal communication and hidden information transport, of content and authorship authentication, and finally of enriching the text with metadata. The watermarking techniques have been explored extensively for multimedia documents in the last decade [1]. In contrast, the studies on natural language watermarking are just starting as attested by the paucity of related papers [2-8, 11-12].

In [3, 4, 6] the techniques of synonym substitution for watermarking have been addressed and various attack scenarios have been described. In [8], Atallah et al. have attempted to use quadratic residues technique to insert a watermark to a given text via synonym substitution. The ambiguity induced on the word precision by the synonym substitution technique has led Topkara et al. to syntax-based natural language watermarking. In a recent study [7], they have considered a syntax-based natural language watermarking. Their technique basically focuses on the syntactic sentence-paraphrasing. In this approach, the raw sentence is parsed by the XTAG parser and then sent for feature verification. Finally, a watermark is embedded into the sentence via transformations on the deep structure of the original sentence.

[1] This work was supported by Boğaziçi Univ. project 05HA203 and TUBITAK project 104E101

In this paper, we investigate the characteristics of an agglutinative language, namely, Turkish, and its relevant syntactic tools for text watermarking. Although watermarking tools, like swapping synonyms, morphological manipulations and punctuation flexibilities can be used for Turkish, it turns out that the syntactic approach offers the most prolific set of text watermarking tools [7]. An original aspect of our study is the consideration of morphosyntactic features such as case, person, tense, aspect, modality and voice features encountered in the original text. This approach is especially relevant since the focus language, Turkish, is an agglutinative language rich in morphosyntactic features to express linguistic notions. We conjecture that the manipulation of the morphosyntactic features has the least negative impact on the semantics of the text when compared with alternative tools such as synonym substitutions and punctuation alterations. Note that Turkish, as an Altaic language with rich morphosyntactic structure, differs significantly from Indo-European languages such as English with respect to a number of structural properties. While languages such as English make use of separate lexical items to express linguistic notions such as temporality, actionality, complex sentence formation (i.e. clausal complements, relative clauses), Turkish makes use of suffixes alongside the separate words for similar purposes. For this reason, we believe that Turkish and other agglutinative [11] and polysynthetic languages provide good ground for syntax based watermarking techniques.

We aim to prove that with syntax-based text watermarking, one can successfully satisfy the three desiderata of natural language watermarking [4], namely:

(i) Semantic equivalence, i.e., there should be negligible difference in meaning between the original text and the marked text. One can expect that syntactic manipulations will cause lesser semantic perturbations as compared to synonym substitutions, especially for a synonym-poor language like Turkish [5]. Furthermore, the syntactic alterations are based on formal descriptions of the linguistic expressions in a sentence domain rather than on the semantic interventions and pragmatic extensions in a word domain.
(ii) Robustness, i.e., watermarks inserted into the text should be robust against malicious or unintentional text alteration. This will be more a property of the watermarking strategy based on a key-instrumented random round-robin method than a consequence of the language.
(iii) Capacity, i.e., one can hope to achieve a target capacity given the rich set of morphosyntactic tools and the possible application of more than one tool per sentence.

This manuscript is organized as follows: In Section 2, we explicate the methodological basis of our study. In Section 3, we describe the Turkish language watermarking model and its performance. Section 4 provides the results of the watermarking experiments and discusses them. In the Conclusion section, we comment on the future studies on the issue.

2. LINGUISTIC TOOLS

2.1 The focus language

We have chosen an agglutinative language as the working language due to the availability of alternative morphosyntactic structures with similar semantic interpretations. Turkish, as an agglutinative language, possesses many advantages in this respect due to its flexible word order and morphosyntactic structure. Earlier studies on watermarking of Turkish texts [5] investigated synonym substitutions, morphological manipulations and punctuation alternations. These studies observed the following difficulties: (i) Synonym substitution is problematic due to the paucity of synonyms in modern Turkish and the fine semantic differences and pragmatic extensions of the lexical pairs, (ii) Low frequency of morphological manipulations (10.7 % in a recent study on watermarking in Turkish texts), and (iii) Unwanted stylistic and meaning differences in punctuation alterations. All these handicaps make the consideration of syntactic tools for watermarking an attractive option. Note that the methodology used in this study can be applied to other agglutinative languages which make use of affixes alongside the separate lexical items to express linguistic notions.

2.2 The syntactic tools

We gleaned twenty one syntactic tools for watermarking from Turkish language. Although the search was not exhaustive, we think that this set represents a fairly complete list of practical tools. This total list was reduced to a subset of seven working tools, the selection being based on two criteria: (i) the frequency of occurrence of the tools in sample corpora, and (ii) the semantic equivalence of the two alternate structures. Some sample tools are: adverb displacement which moves sentential/temporal adverbs to sentence initial or immediately post subject position (e.g., *I will go tomorrow*. vs. *Tomorrow I will go.*), active-passive transformation, swapping the conjuncts in coordinate structures, temporal adjunct clauses, which are expressed through a suffix or a lexical item, necessity expressions formed with a lexical item or a suffix, coordination sentences formed with a suffix or with a lexical item, finite vs. infinite use of embedded clauses, although-clauses formed with different suffixes. The complete list of the selected tools is given in Table I.

Table I: The syntactic tools used for Turkish watermarking (The *frequency* column describes the frequency of the tools in sample corpora and '*' stands for the tools which are not available in English)

Tool	frequency	Example (1st line: Turkish, 2nd line: gloss, 3rd line: English)
1. Active/passive voice	55.52	İşçiler **kumu** taşıdı. / **Kum** işçiler **tarafından** taşı**n**dı Workers sand carried / sand workers by was carried Workers carried the sand. vs. The sand was carried by workers.
2. Adverb displacement	5.41	Ali **yarın** Istanbul'a gidecek. / Ali tomorrow to Istanbul will go **Yarın** Ali İstanbul'a gidecek. tomorrow Ali to İstanbul will go Ali will go to Istanbul tomorrow. vs. Tomorrow Ali will go to Istanbul.
3. Conjunct order change	24.7	**Ali** ve Ayşe / Ayşe ve **Ali** Ali and Ayşe Ayşe and Ali
*4. Verb1 and verb2 / verb+(y)Ip verb2		Ali eve gel**di ve** yattı / Ali eve gel**ip** yattı Ali to home came and slept / Ali to home came and slept Ali has come to home and slept.
*5. Verb-NOUN-POSS when / verb-NOUN-POSS-LOC	2.92	Eve geldiğim **zaman** uyuyordun. / to home I came when you were sleeping Eve geldiğim**de** uyuyordun. to home I came-when you were sleeping You were sleeping when I came to home.
*6.Verb-NOUN1-POSS although1 / verb-NOUN2-POSS-DAT although2		Baktığım **halde** göremedim. / I look-NOUN1 although1 I couldn't see Bakmama **rağmen** göremedim. I look-NOUN2 although2 I couldn't see Although I looked at it, I could not see.
*7. Verb-TENSE-AGR because1 / verb-NOUN-POSS because2	2.63	O eve geç gel**di diye** ona kızdım. / s/he to home late came because1 to her/him I got angry O eve geç gel**diği** **için** ona kızdım s/he to home late come-NOUN because2 to her/him I got angry I got angry because s/he came to home late.
*8. Subject-GEN verb-NOUN-POSS have to / subject-NOM verb-	0,27	Ali'**nin** çok çalış**ması** **gerek.** **/** Ali çok çalış**malı.** Ali-GEN hard study-NOUN necessary / Ali hard must study Ali has to study hard.

NECESSITY		
*9. Verb-PARTICLE be-NOUN-POSS / verb-NOUN-POSS	0,09	Geç gel**miş olması** beni kızdırdı. /Geç gel**mesi** beni kızdırdı. late came be me made angry / late come-Noun me made angry Her/his coming late made me angry.
*10. If verb-COND / verb-COND	0.68	**Eğer** erken gelir**se**, gideriz. / Erken gelir**se** gideriz if early come-COND we go / early come-CON we go If s/he comes early, we will go.
11. Sentence 1: subject-predicate-DIR / Sentence 2: predicate subject-DIR		**Damgalama** önemli çalışma alanlarından biri**dir.** / watermarking important area of study of the one is watermarking is one of the important areas of study. Önemli çalışma alanlarından biri **damgalamadır.** important area of study of the one watermarking is One of the important areas of study is watermarking.
*12. Noun do/ Noun be		Size **yardım ed**ebilir miyim? Size **yardımcı ol**abilir miyim? to you make help can I / for you be helper can I Can I help you?
*13. Noun-without1 /Noun-without2		Toplantıya Ahmet'**siz** başlayabiliriz. / to meeting Ahmet without we can start Toplantıya Ahmet **olmadan** başlayabiliriz. to meeting Ahmet without we can start We can start to the meeting without Ahmet.
*14. Verb1-ArAk / Verb1-A+ verb1-A	4,26	Ali eve koş**arak** geldi. / Ali eve koş**a** **koşa** geldi Ali to home by running came / Ali to home running running came Ali came home by running.
*15. Emotional Verb-NOM-POSS-A according to1/ Emotional Verb-NOM-POSS according to2		Duyduğum**a göre** Ankara'ya gidiyormuşsun. / what I hear according to to Ankara you are going Duyduğum **kadarıyla** Ankara'ya gidiyormuşsun As far as I heard to Ankara you are going According to what I heard, you are going to Ankara.
16. Subject-GEN verb-NOUN-POSS obvious / Obvious that subject-NOM verb-TENSE-AGR	0,53	Bu işin o kadar kolay olmayacağı **belli**. / this work that easy will not be obvious **Belli ki** bu iş o kadar kolay olmayacak. It is obvious that this work that easy will not be That this work will not be so easy is obvious. / It is obvious that this work will not be so easy.
*17. More1 verb-AOR-CON-AGR more2 / verb-DIKçA		**Ne kadar** çalışırsak **o kadar** iş çıkıyor. / more we work more work we have Çalıştıkça iş çıkıyor. As we work more more work we have As we work more, we have more work.
*18. Maybe verb-TENSE-AGR / Verb-POSSIBILITY-TENSE-AGR	9,8	Ali **belki** bu akşam gelir. / Ali bu akşam gel**ebil**ir Ali maybe tonight comes / Ali tonight may come Ali may come tonight.
*19. Verb-NEG-IMP-AGR so that / Verb-NEG-NOUN-POSS because		Yorulmasın **diye** az iş verdim. He will not get tired so that little work I assigned Yorulmaması **için** az iş verdim. He will not get tired because little work I assigned I have assigned little work, so that he will not get tired.
*21. Erasure watermark	25	This is the case where none of the watermarking tools is applicable.

Note that all of these syntactic tools include structural alterations on the original sentence enabled by morphosyntactic transformations on the words. These morphosyntactic features such as case, person and number agreement information on nominal expressions, tense, aspect, modality and voice information on verbal expressions are coded on the lexical categories. For instance, the syntactic tool #5 in Table 1 includes the alteration of the linguistic form [Verb-NOUN-POSS when] with another linguistic from [verb-NOUN-POSS-LOC]. In these linguistic forms, the items in CAPITAL letters correspond to a morphosyntactic feature (i.e. NOUN means a nominalizer suffix which makes nominal items out of verb stems, POSS means a possessive suffix which indicates a person in nominal domains, LOC means a locative suffix which is a case suffix indicating spatial and/or temporal location).

All these tools are bidirectional or half-duplex. In other words, they can be applied one way or in the reverse way yielding binary alternatives. For example, the adverb can precede or succeed the subject. For the simplicity of the discourse, we will declare one direction of the tool as "forward" (say, adverb precedes the subject in Tool #2, and as "backward" (adverb succeeds the subject), although obviously these assignments are totally arbitrary.

In the sequel, we illustrate these syntactic manipulations via tree diagrams where one or more synonymous tree variety is mapped to one logical bit value, while the varieties in the remaining group to the alternate bit value.

2.3 The corpus and Treebank:

The corpus [9] used in this study consists of 5536 sentences from texts of different kinds including short stories, novels and newspaper texts. The input sentences taken from this corpus are in Treebank representation, that is to say, the processed forms of the raw sentences of the texts. Below is one sample sentence in Treebank format.

```
- <S No="1">
  <W IX="1" LEM="" MORPH="" IG="[(1,"Osman+Noun+Prop+A3sg+Pnon+Nom")]"
      REL="[2,1,(SUBJECT)]">Osman</W>
  <W IX="2" LEM="" MORPH="" IG="[(1,"gel+Verb+Pos+Past+A3sg")]"
      REL="[4,1,(SENTENCE)]">geldi</W>
  <W IX="3" LEM="" MORPH="" IG="[(1,"galiba+Adv")]"
      REL="[2,1,(MODIFIER)]">galiba</W>
  <W IX="4" LEM="" MORPH="" IG="[(1,".+Punc")]" REL="[,( )]">.</W>
    </S>
```

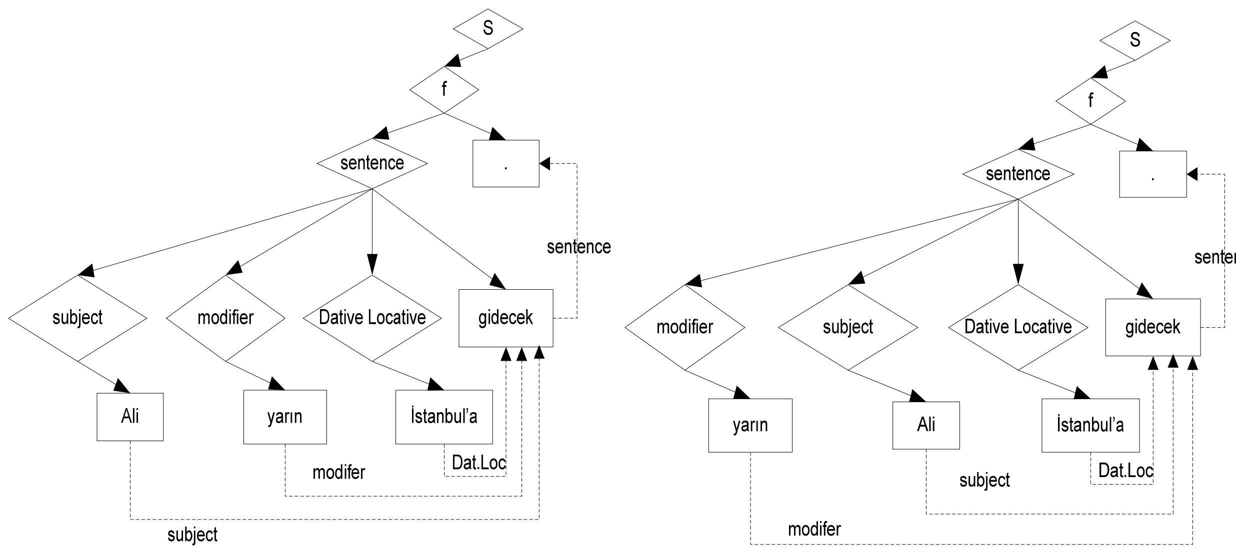
Fig. 1. Treebank representation of a sentence

Sentence: Osman geldi galiba.
Osman came it seems.
"It seems that Osman has arrived"

In this representation, each <W…> word corresponds to a word, where punctuations are also treated as such. The words are given in their morphological structure where each morpheme corresponds to a grammatical function such as voice, tense, person etc. A REL= line corresponds to the sentential function carried by each word in the representation. In other words, the smaller parts (i.e. clauses and phrases) in the sentence are organized/connected to each other via the sentential functions given in REL= line [13-15].

The above dependency parsing is instrumental in the use of many syntactic tools, more specifically, for tools other than #2, #3 and #4 in Table 1. On the other hand, the tools #2, #3 and #4 necessitate a deeper understanding of the syntax. Notably, organization of phrases and clauses functioning as complex subjects and complement clauses via functional dependencies are problematic. An example for case 1 is clausal subjects which include sentences (*That he did not make any progress on the project* made me angry), and an instance for case 2 is clausal objects which include sentences (I know *that he has been working on the issue for 10 years*). We overcame these problems by transforming the linear functional dependencies in [9] to hierarchical relationships. To this effect, we have developed a software tool which converts the Treebank format into a syntactic tree structure which includes syntactic hierarchies. This tool clones each word node and creates its parents as functional nodes, by setting their functions to the same relation as their associated words. Subsequently, it traverses through functional nodes and appends each active node to the related node as its child. All functional nodes that have no relation to another functional node are directly appended to the sentence root.

The algorithm above works with nearly 92% success rate, where sentences are fully represented both in correct order (reading order of word leaves from left to right) and in hierarchical position. Since some dependency links cross each other and the algorithm above cannot handle them in correct order, referential dependency links that indicate the order of a word in sentence are not removed from the representation. The rest 8% of the sentences have been manually corrected. The following Figure 2 exemplifies the tree representations of the syntactic tool #2 in Table 1.



Original sentence tree	*Derived sentence tree*
Ali yarın Istanbul'a gidecek	Yarın Ali Istanbul'a gidecek.
Ali tomorrow to Istanbul will go	tomorrow Ali to Istanbul will go
Ali will go to Istanbul tomorrow	Tomorrow Ali will go to Istanbul.

Figure 2: Tree structure of a sentence before and after adverb displacement transformation

Figure 2 shows tree structures of a sentence representing the phrases and clauses of a sentence in a hierarchical way while respecting their functional dependencies.

3. WATERMARK EMBEDDING and DETECTION

The rationale of the watermarking method is to weave through the text by applying feasible syntactic watermarking tools on each occurrence. The details of the watermarking algorithm at the sentence level and text level are described below.

3.1 Sentence–level preprocessor for watermarking

The preprocessing of the input text for natural language watermarking is described in Figure 3. In this diagram, the curly rectangles denote the text in its various stages of processing while the pure rectangles stand for operators.

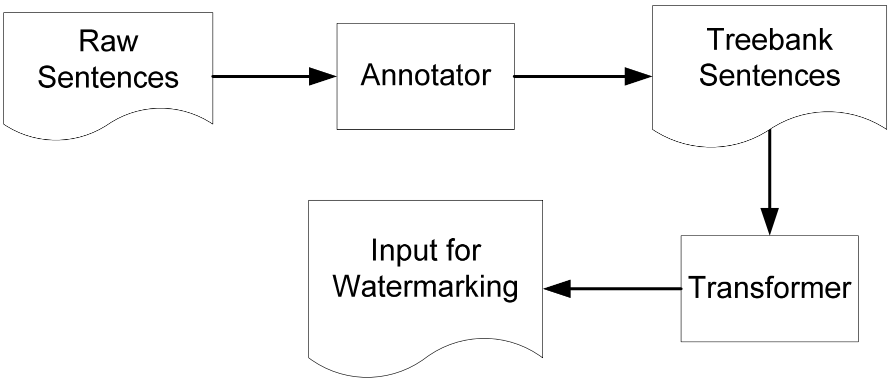


Fig. 3: Preprocessing of the text for sentence level watermarking

The two operators in the square blocks of the diagram are:

- Annotator: The annotator [9] processes the raw sentences with morphosyntactic features and functional dependencies. It parses the sentences into smaller categories. First, the morphosyntactic features (i.e. case features of nouns, tense and person features of verbs) are attached to the lexical items, such as words. Then, the functional dependencies of the words, such as subject, modifier and verb, are marked. This procedure forms the Treebank representations of the sentences (Fig. 1).
- Tree Transformer: The parsed and annotated sentences are operated by the transformer, which converts the Treebank sentences into a syntactic tree representation. The transformer uses the constituent information of the words and matches this constituent information with the functional dependencies of the lexical items. Figure 4 illustrates the conversion of a word in an input sentence for text watermarking.

Raw word:	geldi (s/he came)	
Annotator:	Adding morphosyntactic features	**[(1,"gel+Verb+Pos+Past+A3sg")]**
	Adding functional dependencies	**[4,1,(SENTENCE)]**">**geldi**</W>
Transformer: Creating a place for the word in the tree representation		

Fig. 4: The conversion of a word via the annotator and the transformer

3.2 Text–level watermarking embedding

Once an input text is rendered amenable for text watermarking, then the syntactic tools can be applied according to a regime for watermark embedding. For security reasons, the order of application of syntactic tools is randomized. We describe the text watermarker by means of three operators, namely, 'watermark tester', 'watermark randomizer' and 'watermark embedder' (Figure 5).

Watermark Tester: Given the tree-structured parsed sentence, the algorithm checks each one of the watermarking tools as for their potential applicability to the sentence. The tools that are checked constitute the pool from which one or more tools will be selected to watermark that sentence. For example, the sentence "Çok yedim." (I ate a lot) does not allow any of the tools, while the sentence "Ayşe ve Ali dün eşyaları taşıdığı zaman ben eve geldim ve kilidi değiştirdim." (Yesterday, when Ayşe and Ali were carrying the staff I came home and changed the lock) allows the application of as much as 4 tools in Table 1 (namely adverb displacement, conjunct order change, Verb1 and verb2 / verb+(y)Ip verb2 and Verb-NOUN-POSS when / verb-NOUN-POSS-LOC).

- Watermark Randomizer: The algorithm chooses one of the available tools in some random order using a secret key shared between the encoder and decoder sites. Alternative approaches would be to select the tools in a round robin fashion or to select them randomly or to select them with weighted probabilities to balance their occurrences. Notice that the randomizer selects tools from the pool of applicable ones for that sentence.
- Watermark embedder: The chosen tool is operated if the watermark message bit to be embedded and the condition of the sentence do not agree; else, there is no change made in the sentence. Assume that the tool chosen is the tool #2, that is, "Adverb displacement tool" or the adverb position vis-à-vis the subject. This adverb rule is as follows: adverb preceding the subject is "1" and adverb succeeding the subject is "0". Now suppose that the message bit is "0": if the actual sentence has already its adverb after the subject, then no action is taken; otherwise the subject and the adverb swap their positions.

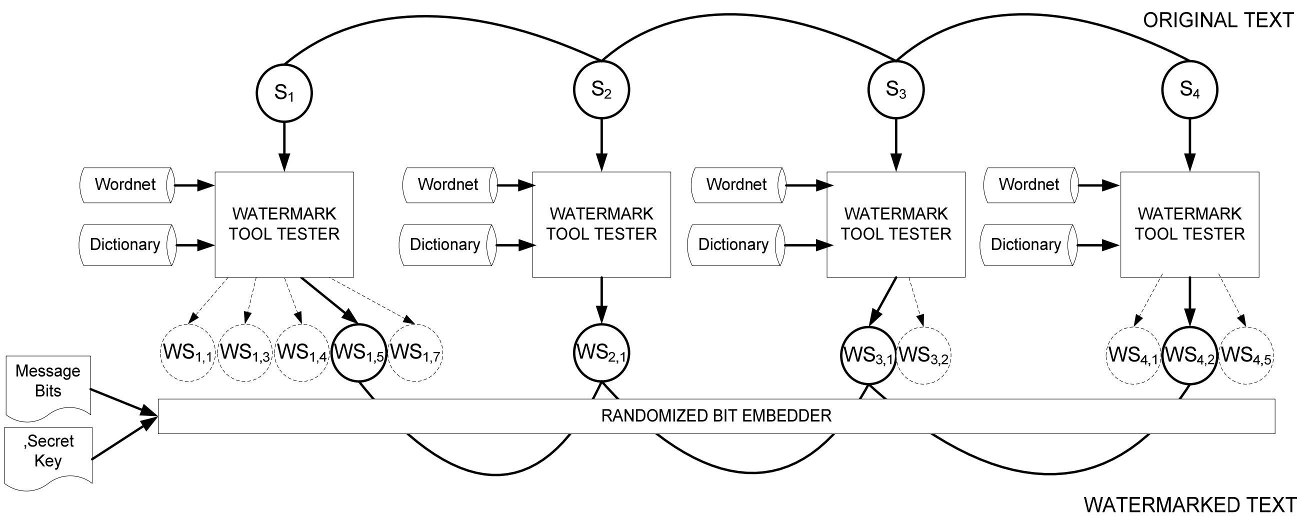


Fig. 5. The watermarking algorithm at the text level: S_1, S_2, S_3, S_4 stand for the sequel of input sentences. The watermark tester checks and lists the tools applicable for sentences. For example, S_1 possesses the list of $WS_{1.1}$, $WS_{1.3}$, $WS_{1.4}$, $WS_{1.5}$, $WS_{1.7}$ tools. The watermark randomizer algorithm chooses one of these tools (say, $WS_{1.5}$). Finally, the embedder watermarks that sentence (0-1 bit values) to the sentences. This procedure is

repeated for each sentence and the chosen forms of the sentences, the tree varieties are connected to each other as watermarked text ($WS_{1.5}$-$WS_{2.1}$-$WS_{3.1}$-$WS_{4.2}$ in Fig. 5).

Implicit in the depiction of the algorithm in Fig 5 are the following: i) The syntactic and semantic properties of Turkish (via Wordnet and Dictionary); ii) Incorporation of the semantic and pragmatic extensions of syntactic tools in a rule based schema. These include exceptions of the morphosyntactic rules in order to avoid their ungrammatical or unconventional applications.

The whole watermarking procedure is illustrated in Figure 6. The preprocessed sentences in tree representation are inputted to the watermarking algorithm. The syntactic manipulations are effected as follows.

1. Each sentence is checked for all morphosyntactic tools and a list is prepared. Note that the application of a syntactic tool is subject to semantic restrictions of lexical items. In other words, semantic properties of lexical items are considered before allowing the application of a syntactic tool. The identification of the semantic properties of the lexical item is enabled by Wordnet [10] For example, passivization makes sense only when the sentence subject is animate for transitive verbs.
2. A tool is randomly selected from the list according to a secret key and operated on the sentence. Note that all tools are bi-directional in order to encode binary data (e.g., active = 1 and passive =0). No change occurs if the bit value and the sentence condition match; otherwise the watermarking change takes place. Note that our procedure differs from the one in [7], where passivization and activization are considered as two separate tools each of which applies in single direction.

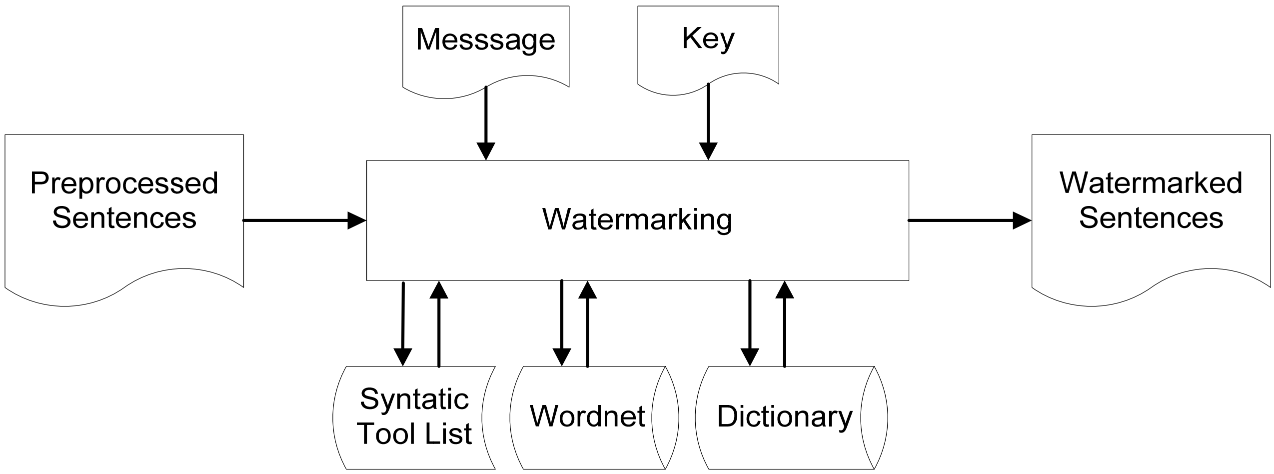


Fig. 6 Syntactic sentence-level watermarking

Security aspects:
The security of the watermarked text is enhanced by both randomization and the injection of a "pass" tool. First, as the text is woven with watermarking modifications as in Fig. 3, there is no apparent fixed pattern of watermarking. This is necessary both to cause the least "damage" to the style and to hide the presence of watermarking, if any, from the adversary. Second a "pass" or "escape code" is injected into the list. This gives the option of not using any watermarking tool despite their availability at a sentence level. In [7], this option was enabled by the insertion of fixed synonym substitutions, while in our algorithm there is no need to signal the occurrence of the void case. We emphasize again that the void case is essential for the obfuscation of the

watermark. Furthermore, we can control the sparseness of the watermarking at the sentence level by adjusting the frequency of the void case. This would force the adversary to do massive re-watermarking in order to corrupt the hidden message.

The text watermarking encoder functions as follows:

Step 1: Input the test sentence to the syntactic parser whose output is a dependency chain,
Step 2: Transform the dependency chain of the sentence to yield a parsed syntactic tree
Step 3: Check out the availability of each tool and select "randomly" one tool from the pool
Step 4: Implement the tool: if the watermark message bit agrees with the condition of the sentence with respect to that tool, then leave the sentence unchanged; else, apply the syntactic alteration.

Similarly, the watermark extractor functions as follows:

Step 1: Input the possibly marked sentence and run the syntactic parser,
Step 2: Obtain the parsed syntactic tree
Step 3: Estimate the pool of the potentially applicable tools and determine the one(s) that must have been applied according to the randomizing scheme
Step 4: Check the direction of the tool: if the tool was applied in a forward manner, then decide for "1"; if it was applied in the backward sense, then decide for "0".

4. DISCUSSION of WATERMARKING PERFORMANCE

In this section, we give the main results of the watermarking experiments and discuss the implications of these results with respect to the natural language watermarking phenomenon.

4.1 Watermarking performance measures

The outcome of the Turkish text watermarking has been evaluated based on two success measures: objective success and subjective success. We have used two texts, containing 154 and 60 sentences respectively to test the success rate of the proposed watermarking method. We define the following concepts of success:

Objective rate: This is an objective figure and denotes the number of times that tool was a viable tool. Note that a tool becomes viable if it is grammatically applicable and can surmount all stylistic and semantic constraints that impede its usage. We give the occurrence rate as a percentage figure computed from the analysis of 5534 sentences.

Subjective rate: Syntactic tools may still perturb the naturalness and style of the text. One way to measure the palatability of the watermarking is to let subjects evaluate the texts and show their reactions by editing attempts. The subjects are given marked texts and asked to edit them for improved intelligibility and style. This is a blind test because the subjects are not aware that text watermarking has taken place. An edit distance, similar to Levensthein edit distance [16], is used where substitutions, insertions and deletions are weighted. The editing actions hitting the watermarking targets can thus be measured, thus indicating the degree of acceptability of the marked text [17].

4.2 Watermarking results

Table II gives the results of the watermarking manipulations on sample texts. The second (Occurrence) column lists the occurrence probability per sentence, also called the "objective success". These frequencies are estimated using a corpus of 5534 sentences. Notice that the occurrence probability of morphosyntactic tools is quite lopsided. For example, the two principle tools of conjunct order change and active-passivization constitute already 80,22 % of the occurrences. To mitigate this lopsidedness of the tool frequencies, one can use weighted fair queuing, such that the more sporadic ones are given higher chance of being used. It is noteworthy to remark that close to one third of sentences, typically short ones, do not admit any watermarking. The "average success" at the bottom of the table gives the sum of the occurrence rates of the six watermarking tools. This figure means that on the average one can expect 0,92 tool to be available per sentence.

The third (Edit) column shows the editing attempts for each tool, that is, the percentage of time that readers felt the urge to modify and correct the location of the sentence where a specific watermarking tool was operated. Obviously, lack of editing effort implies imperceptibility of the tool's operation.

Table II: The occurrence frequency and success rate of syntactic watermarking tools

No.	Tool	Occurrence %	Edit %
1	Active/passive voice	55,52	29,1
2	Adverb displacement	5,41	11,6
3	Conjunct order change	24,7	7,9
4	Verb-NOM-POSS when / verb-NOM-POSS-LOC	2,92	1
5	Verb-TENSE-AGR because1 / verb-NOM-POSS because2 – verb-NOM-POSS-ABL because3	2,63	6,3
6	If…verb-COND / verb-COND	0,23	0,0
7	Erasure watermark	25	3,9
	Total/Average	91,41	12,7

A low value in this last column of Table II indicates the imperceptibility of the tool application. Note that this imperceptibility results from stylistic effects rather than grammaticality vs. ungrammaticality or semantic anomaly of the sentences. One can observe that the active-passive tool, while occurring the most frequently, also attracts the highest edit reactions. This suggests that this tool must be applied with parsimony. It is also interesting to note that sentences which include erasure watermarks (sentences which have not been transformed) have also received edit hits at a rate of 3,9 %, implying that stylistic editing is not a crucial problem for text watermarking.

CONCLUSION

In this paper, we investigated syntactic tools for natural language watermarking of Turkish.. In line with [7], we have shown that the extensive repertoire of morphosyntactic tools of Turkish provide a fertile ground for natural language watermarking without disturbing the semantics and stylistics of texts. The text-watermarking algorithm works by transforming raw sentences into their Treebank representation and then into their syntactic tree. A software tool weaves through the text by checking the applicable tools and randomizing their occurrence.

Future work will focus first on watermarking adaptation to the type or genre of texts. A more complete repertoire of watermarking tools will be implemented and the embedding capacity determined experimentally and checked against statistical predictions. . These efforts will be accompanied by the development of a new parser for Turkish sentences, which embeds both hierarchical relations and functional dependencies into the raw sentences. Another direction for the study is to use automatic translation software as in [7] to measure the success rate of each tool. It would be interesting to compare the native speaker edit reactions, as a performance measure, against that of automatic translators. One can envision developing a stylistic watchdog software, which would consider the text as a whole and apply stylistic semaphore and/or corrections. This will take into account the correlations between watermarking transformations. For example, adverb displacement is neutral enough that it can be applied anytime available. On the other hand, if a sentence is active and the following one is transformed to passive, then the present one must also be changed to passive. Finally, robustness of the watermarking scheme against attacks and synchronization aspects remains to be explored.

REFERENCES

1. I. Cox, M. L. Miller, J. A. Bloom, M. Kaufman, Digital Watermarking, 2002.
2. M. S. Khankhalli, K. F. Hau, Watermarking of Electronic Text Documents. *Electronic Commerce Research,* 2, 169-187, 2002.
3. M. Topkara, C. M. Taskiran, E. J. Delp, Natural language watermarking. *SPIE Conf. On Security, Steganography and Watermarking of Multimedia Contents VI*, San Jose 2005.
4. C. M. Taskiran M. Topkara, E. J. Delp, Attacks on linguistic steganography systems using text analysis. *In SPIE Conf. On Security, Steganography and Watermarking of Multimedia Contents VII,* San Jose 2006. no: 3&4, pp. 313-336, 2006.
5. H. M. Meral, B. Sankur, A. S. Özsoy, Watermarking tools for Turkish texts. *In Proceedings of National Conference on Signal Processing and Application (SIU 2006).* Sabancı University, Antalya 2006.
6. U. Topkara, M. Topkara, M. J. Atallah, The Hiding Virtues of Ambiguity: Quantifiably Resilient Watermarking of Natural Language Text through Synonym Substitutions. *In Proceedings of ACM Multimedia and Security Conference*, Geneva, 2006.
7. M. Topkara, U. Topkara, M. J. Atallah, Words are not enough: Sentence level natural language watermarking, *MCPS'06*, Santa Barbara 2006.
8. M. J. Atallah, C. J. McDonough, S. Nirenburg, V. Raskin, Natural language processing for information assurance and security: an overview and implementations. *In Proceedings of 9th ACM/SIGSAC New security paradigm workshop*, pp. 51-65, Cork, Ireland, 2000.
9. K. Oflazer, B. Say, D. Z. Hakkani-Tür, G. Tür, Building a Turkish Treebank, In Building and Exploiting Syntactically-annotated Corpora, Anne Abeille (ed.), Kluwer Academic Publishers, 2003.
10. O. Bilgin, Ö. Çetinoğlu, K. Oflazer, Building a Wordnet for Turkish. In Romanian Journal of Information Science and Technology, Vol.7, no. 1-2, 163-172, 2004.
11. O. Takizawa, K. Makino, T. Matsumoto, H. Nakagawa, I. Murase, Method of hiding information in Agglutinative language documents using adjustment to new line position. In R. Khosla (eds.), *KES 2005, LNAI 3683*, pp. 1039-1048, 2005.
12. Y, Liu, X. Sun, Y, Wu. A natural language watermarking based on Chinese syntax. In Advances in Natural Computation. pp. 958-961, Springer, 2005.
13. G. Eryiğit, J. Nivre, K. Oflazer, The incremental use of morphological information and lexicalization in data-driven dependency parsing. Ms. Istanbul Technical University.
14. G. Eryiğit, K. Oflazer, Statistical dependency parsing of Turkish. In proceedings of EACL 2006, pp. 89-96.
15. K. Oflazer, dependency parsing with an extended finite state approach. Computational linguistics, vol. 29:4, 2003.
16. K. Lowett, Levenstein edit distance. http://www.call-with-current-continuation.org/eggs/levenshtein.html
17. M. Snover, B Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, A study of translation edit rate with targeted human annotation, Proceedings of Association for Machine Translation in the Americas, 2006.