Generation of Noun Compounds in Hebrew: Can Syntactic Knowledge be Fully Encapsulated?

Yael Dahan Netzer and Michael Elhadad
Ben Gurion University
Department of Mathematics and Computer Science, Beer Sheva, 84105, Israel
(yaeln|elhadad)@cs.bgu.ac.il

Abstract

Hebrew includes a very productive noun-compounding construction called *smixut*. Because smixut is marked morphologically and is restricted by many syntactic constraints, it has been the focus of many descriptive studies in Hebrew grammar.

We present the treatment of smixut in HUGG, a FUF-based syntactic realization system capable of producing complex noun phrases in Hebrew. We contrast the treatment of smixut with noun-compounding in English and illustrate the potential for paraphrasing it introduces.

We specifically address the issue of determining when a smixut construction can be generated as opposed to other semantically equivalent constructs. We investigate several competing hypotheses – smixut is lexically, semantically and/or pragmatically determined. For each hypothesis, we explain why the decision to produce a smixut construction cannot be reduced to a computation over features produced by an outside module that would not need to know about the smixut phenomenon.

We conclude that smixut provides yet another theoretical example where the interface that a syntactic realization component presents to the other components of a generation architecture cannot be made as isolated as we would hope. While the syntactic constraints on smixut are encapsulated within HUGG, the input specification language to HUGG must contain a feature that specifies that smixut is requested if possible.

However, because smixut accounts for close to half the cases of NP modifiers observed on a corpus of complex NPs, and it appears to be the unmarked realization form for some frequent semantic relations, we empirically evaluate a default setting strategy for the feature `use-smixut` based on a simple semantic classification of the relations head-modifier in the NP. This study provides a solid ground for the definition of a small set of predicates in the input specification language to HUGG, that has applications beyond the selection of smixut — for the determination of the order of modifiers in the NP and the use of stacking vs. conjunction — and for the definition of a bilingual input specification language.

1 Introduction

Over the past three years, we have started developing HUGG, a syntactic realization component for Hebrew. One of our objectives is to investigate constraints on the design of the input specification language to a syntactic realization component through a contrastive analysis of the requirements of English and Hebrew. By design, we are attempting to keep the input to HUGG as similar as possible to the one we defined in the SURGE syntactic realization for English [8]. A detailled analysis of syntactic constructs specific to Hebrew becomes, therefore, critical to evaluate to which extent the input specification language can abstract away from knowledge of the syntax.

We investigate in this paper one such construct: the Hebrew noun-compounding form known as *smixut*. Because smixut is morphologically marked and remarkably productive in Hebrew, there exists a vast tradition of work in descriptive grammar of Hebrew providing functional analysis of the phenomenon [12] [11] [15]. This previous work has served as a fertile ground for our own generation-specific purposes.

The specific issue we discuss in this paper is: what information in the input specification to the syntactic realization component can license the selection of a smixut construct. The classical objectives of modularity and knowledge encapsulation indicate that this decision should be a private decision of the syntactic realization component. Because there are so many syntactic constraints on the use of smixut, the objective of encapsulation is made even more desirable.

After a thorough analysis of the different functions of the smixut construct and the constraints over its use, our conclusion, however, is that this reductionist strategy fails: we cannot explain the selection of a smixut construct without considering simultaneously lexical, semantic and pragmatic factors.

Theoretically, in order to allow the syntactic realization component to select a smixut construct adequately, we are, therefore, left with two options: (1) either provide full, detailed access from the syntactic realization component to the complex semantic and pragmatic features that can impact on the decision; or else, (2) allow the other components to request the use of a smixut construct when they deem it adequate. In either case, modularity and encapsulation suffer. This analysis informs us in our design of a bilingual realization component: if a feature like `use-smixut` is required in the input to the syntactic component, this level of abstraction cannot be appropriate as a bilingual construction. It also informs us in the general ongoing debate over the design of reusable syntactic components and their place in the architecture of generators.

From a more pragmatic perspective, however, we also provide a set of simple defaults for the generation of smixut based on a simple semantic classification of the relations head-modifier. We evaluate the validity of this classification by constructing input specifications for a corpus of more than 800 complex noun phrases and regenerating from them. The validation process includes two aspects: (1) we test that human coders agree on the semantic relations they use to label complex NPs; and (2) we verify that the generator's decision to produce a smixut construction corresponds to that observed in the corpus. Preliminary results are provided in Section 4.3. They encourage us to view in the set of semantic relations we propose a useful basis for the design of an interlingual input specification language.

In the rest of the paper, we first briefly review the main approaches to the treatment of noun-compounds in English and in Hebrew. In Section 3 we provide descriptive data on the use of smixut in Hebrew. We then describe in Section 4 a first approach to the generation of smixut based on a simple semantic classification similar to that found in [13]. In Section 5, we identify the limitations of such an approach, illustrating that an explanation based on recoverable semantic relations cannot provide sufficient nor necessary conditions for the generation of smixut. However, the preliminary empirical evaluation we present in Section 4.3 demonstrates that the semantic relation approach provides a useful default that works "most of the time."

2 Previous Work

2.1 Noun compounds in English

Noun compounds in English are partly "frozen" lexical constructions (*e.g.*, *computer science*) and partly compositional constructions (*e.g.*, *computer equipment, farm equipment, city equipment...*).

The problematic aspect of this construction is that it seems to be very productive in English, but yet severely constrained (*e.g.*, * *science equipment*). Compound constructions are also regularly ambiguous.

The various approaches developed to explain the construction of noun compounds and their interpretation can be classified in three groups: semantic, pragmatic, and statistical/lexical.

Semantic theories explain the production of a noun compound *N1 N2* as a derivation from a semantic relation *N1 R N2* where the relation *R* is elided. The theory of recoverably deletable predicates (RDPs) of [13] proposes that only a small set of relations (cause, have, make...) can participate in this process. Because these relations were too general and sometimes vague, and because one can observe many cases of compounds that do not correspond to any of the proposed RDPs, others have proposed to define more precise domain specific models to explain the deletion of certain relations.

Recognizing the importance of contextual factors, pragmatic theories predict the use of noun compounding when relations like naming or constrast play a role [7]. For example, when refering to two persons wearing a jacket and a coat respectively, one can use compounds like *the jacket man* and *the coat man* even though, in neutral contexts, it would be difficult to interpret the same compound (*i.e.*, the *wear* relation is not deletable).

In [6], the explanation for compounding is provided in the form of lexical/syntactic knowledge. Generative devices inspired by [17] are found in the lexicon. In addition, statistical knowledge predicts which derivations are the most likely.

From a generation perspective, the problem is less acute than for interpretation: we must decide whether to construct a compound as opposed to recover the missing relation between the head and the modifier. The problem has, therefore, not received heavy attention for English generation. In the past, we used Levi's model in generation [9], but as part of the lexical chooser, and we did not include it within the syntactic realization component.

In Hebrew, however, the smixut construction is extremely productive (in our corpus, smixut modification accounts for 40% of all modifiers, more than any other type of syntactic modification in NPs). We, therefore, had to address the issue of when to generate smixut as a priority in the development of the NP grammar for HUGG.

2.2 Noun compounds in Hebrew

The structure of noun compounds in Hebrew – smixut, is marked and, therefore, it has been the focus of Hebrew language studies. The head is marked morphologically and it does not carry a mark of definiteness even though it is semantically definite. [11] [16] provide detailed studies of the syntactic constraints on the use of smixut. We provide an overview of the main constraints in Section 3.

Although smixut is traditionally treated as a possessive construction, it can express many other relations between head and modifier. Levi [12] has extended her treatment of the noun-noun relation in English [13] and proposed that the same semantic relations can all be expressed by the Hebrew smixut construction. [2] and [11] (Chapter 6) also provide similar semantic classifications of the elided relation in a smixut. We build on these studies in our implementation, but also investigate how a semantic account can be integrated with pragmatic and lexical constraints.

3 Noun Compounds in Hebrew: Constraints

We briefly present in this section the basic syntactic constraints over the use of smixut in Hebrew. The notion of "smixut" covers three main constructions: [2] [11][p24]:

compound	**separate construct**	**double-genitive**
cadur ha-tinok	*ha-cadur Sel ha-tinok*	*cadur-o Sel ha-tinok*
ball the-baby	*the ball of the baby*	*ball-his of the-baby*

Smixut is identified by two main tests: first, when plural is used, only the head is marked morphologically with a special inflection: *yeled* (child - singular) vs. *yelad-im* (children - plural non-smixut) vs. *yald-ei* (children - plural head of smixut marking).

Second, when definite is used, only the modifier is marked even though the head is understood as definite: *aron mitbax* (cabinet kitchen) (a kitchen cabinet) vs. *aron ha-mitbax* (cabinet the-kitchen) (the kitchen cabinet).

Gerunds are also built using a smixut construction around the infinitive form of the verb.

The arrival of the doctor:	*Bo ha-rofe*	*Bo'o Sel ha-rofe*
	Arrive the-doctor	*Arrive-his of the-doctor*

3.1 Syntactic Constraints

One of the main constraints on the use of smixut is that a head can have only one modifier in somex position. When several modifiers are attached to a head, this constraint forces other relations to be realized in other syntactic constructions (pre-modifier adjective or post-modifier prepositional phrase or relative clause). For example, when refering to a suit made of leather, the default realization (unmarked) is the smixut *beged wor (suit-leather)*. The alternative realization *beged me-wor (suit from-leather)* with a qualifier PP is also possible, but less frequent. However, when referring to a bathing-suit in leather[1] the default realization is *beged-yam me-wor (suit-sea from-leather)*. Because the somex position of the head *beged* is occupied by the *yam (sea)* modifier, the second modifier (leather) is relegated to a non-somex position.

The head of a smixut must be a noun or a conjunction of nouns and it cannot be a compound itself. This means that smixut only allows right branching[2]. This is in contrast with English, which allows right or left branching constructions: *(computer communication) system* vs. *computer (communication system)*.

Pronouns and proper nouns cannot head a smixut, and any pronoun in the modifier position (possible in possessive relations, for instance) has the objective case and is agglutinated to the noun: *ben-o son-him (his son)*.

There are several restrictions on the combination of smixut with different determiner types. Noun phrases in Hebrew are polydefinite — that is, definiteness is marked on several of the constituents in the phrase. Any adjectival modifier is marked with agglutinated definite markers, the same as the head noun. Quantifiers and determiners can be also marked. In smixut, only the mod-n is marked as definite. Therefore, compounded nouns are understood as having the same definitness value. As a consequence, if definiteness of the head and modifier differs, smixut cannot be used:

[1] Very frequent on Israeli beaches.

[2] That is, if Hebrew is written left to right.

head-N + mod-N definite (The son of the king)	*ben ha-melex* *son the-king*	*ben-o Sel ha-melex* *son-his of the-king*
head-N + mod-N indefinite (a son of a king)	*ben melex* *son king*	*ben Sel melex* *son of king*
head-N indefinite, mod-N definite (a son of the king)	*ben Sel ha-melex* *son of the king*	
head-N definite, mod-N indefinite (The son of a king)	*ben-o Sel melex* *son-his of king*	

3.2 Lexical Constraints

Not every noun can head a smixut construction: words which are lexical-compounds (*cadursal - ball-basket - basketballl*), words of foreign origin, cannot be in nismax form, and thereofre any modifiers must be realized in another syntactic construction.

Several criteria exist to distinguish frozen from productive smixut compounds: frozen compounds behave like regular smixut with respect to plural marking (special morphology inflection). But depending on the level of cohesiveness of the frozen compound, definite marking may differ: *beyt-sefer – house-book – (a school)* may give *ha-beyt-sefer (the school)* instead of the predicted *beyt ha-sefer* for a productive smixut.

In addition, for frozen constructs, many additional constraints exist: the nismax cannot be modified (** beyt sefer kri'a – house book reading*), cannot change its number (** beyt sfar-ym – house books*), cannot be taken apart (** bayt Sel sefer – house of book*) or be conjoined to another somex (*beyt sefer ve beyt Hol-im – house book and house patient-s – a school and hospital* but ** beyt sefer ve-Hol-im – house book and patient-s*). Detailed references from linguistic and sociolinguistic aspects are found in [5] and [4] respectively.

3.3 Semantic Constraints

Smixut is often understood as a genitive type of construct, expressing dominantly a possessive relation between the head and the modifier. Very often, however, the relation expressed is not one of possession.

The semantic relation realized by the smixut has an influence on the possible paraphrases the smixut can receive [3]: some semantic relations (including possessive) can be realized in a double genitive construction while others can only be realized by a simple smixut. The semantic relation also determines which types of modifiers can be accepted in the smixut construction. In general, when a double genitive construction is not possible, then pronouns cannot appear as modifiers, even in a simple smixut.

In contrast, in the case of gerunds, the only possible structures are compound and double-genitive while the separate construction is not possible.

Levi [12] claims that smixut realizes in Hebrew a number of universal semantic processes which exist in other languages, thus extending her original analysis for English [13]. Her "non predicative modifiers" theory claims that Noun-noun compounding is produced by two syntactic processes: nominalization or deletion of the predicate – which corresponds to the observed uses of smixut for possessive and gerund construnctions.

Azar [2] classifies smixut into 15 semantic categories. This classification can be made parallel to Levi's RDPs. Glinert [11] also refers to such a classification in a similar manner.

3.4 Pragmatic Constraints

There are cases, however, when smixut can be constructed with no regard to The semantic set that was identified. Certain contexts license smixut constructions that would not be obtained otherwise – for example, contrast or naming [7].

In addition, smixut is associated with style and genre parameters. Seikevicz [19] analyzes transcripts of spoken Hebrew and compares smixut usage in spoken vs. written Hebrew, finding a marked difference.

Other pragmatic considerations for the use of smixut include the objective to generate a more compact text and to make of a compound an item available for further anaphorical reference.

Finally, decision to compound a head with a plural or singular modifier is related to the genericity of the description and to the habituality of the relation, as is the case in English [18] (p. 916):

The table in the corner was laid for dinner	*The corner table*
The girl in the corner spoke to me	* *The corner girl*

4 When Can Smixut Be Generated?

Our main objective is to determine what features must be present in the input to the syntactic realization component to decide when to use a smixut construction.

We observe that the production of smixut is semantically constrained, and that the semantic relation holding between head and modifiers determines which syntactic paraphrases are possible (among smixut, double-genitive and separate construct). A set of semantic predicates similar to Levi's RDPs seems to play a role in the decision. On the other hand, being a member of that set is not a sufficient nor necessary condition to generate a smixut.

In the SURGE grammar for English, we did not address this decision, and assume that the input includes a predefined syntactic construction (classifier-head). For Hebrew, we must find an alternative approach because: (1) smixut is extremely frequent (40% of the noun modifiers in our corpus); (2) smixut is the default realization for many relations but it cannot be used in many syntactic contexts.

4.1 Exploiting a Semantic Classification

Our strategy is to provide in the input to HUGG a reliable default indicating that smixut should be used when possible, but making it possible to fall back on an alternative realization (separate or double genitive, or qualifier modification) when smixut is not possible.

For instance, lexical-compounds cannot be head a smixut, and, therefore, their modifier must be realized as a PP. The same semantic relation (*e.g.*, material) will be realized in two different ways depending on the lexical property of the head:

mewyl wor
coat leather
leather coat

$$\begin{bmatrix} \text{cat} & \text{np} \\ \text{head} & \begin{bmatrix} \text{lex} & \text{“mewyl”} \end{bmatrix} \\ \text{modifiers} & \begin{bmatrix} \text{material} & \begin{bmatrix} \text{lex} & \text{“wor”} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

If the same input is provided, but the property of the head noun is different, a different construction will be generated:

beged-yam me-wor
bathing-suit from-leather
leather bathing-suit

$$\begin{bmatrix} \text{cat} & \text{np} \\ \text{head} & \begin{bmatrix} \text{lex} & \text{“beged-yam”} \\ \text{cat} & \text{noun-compound} \end{bmatrix} \\ \text{modifiers} & \begin{bmatrix} \text{material} & \begin{bmatrix} \text{lex} & \text{“wor”} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

A similar mechanism would determine that a smixut is not possible if the definiteness of the head and the modifier do not match, as discussed above in the *a son of the king* example.

The syntactic realizer also relies on the semantic classification of the relation Head-modifier when several modifiers are attached to a single head. In this case, only a single modifier can be realized as a smixut. The others must be realized differently. In this case, the realizer must determine which relation takes priority to become the smixut, and it must also provide an appropriate paraphrase for the non-smixut modifier.

For example, the English NP *leather house shoe* will be generated in one of the following ways:[3]

nawal bayit me-wor
house shoe from-leather

nawal wor la-bayit
shoe leather for-the-house

$$\begin{bmatrix} \text{cat} & \text{common} \\ \text{lex} & \text{“nawal”} \\ \text{modifiers} & \begin{bmatrix} \text{material} & \begin{bmatrix} \text{cat} & \text{common} \\ \text{lex} & \text{“leather”} \end{bmatrix} \\ \text{purpose} & \begin{bmatrix} \text{lex} & \text{“bayit”} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Beyond smixut-related decisions, determining a set of semantic relations is also useful to allow HUGG to determine appropriate defaults for prepositions in PP modifiers. For example, in the example above, HUGG can select the default preposition *for* in *shoe for the house* because the relation of purpose is specified.

The same classification is also useful to determine the order and the syntactic structure of a multi-modifier sequence in complex NPs.

In general, when several modifiers attach to a single head, a broken (conjuncted) sequence is created [10] (in contrast to English, where a stacking construction is generally used): *A big white house* vs. *bayit gadol ve-lavan (house big and white)*.

However, when adjectives realize a semantic relation that could have been realized by a smixut, they appear first in the sequence of modifiers and they do not require a conjunction [1].

makdeHa HaSmalyt gadola
driller electronic large
a large electronic drill

$$\begin{bmatrix} \text{cat} & \text{common} \\ \text{lex} & \text{“makdeHa”} \\ \text{modifiers} & \begin{bmatrix} \text{instrument} & \begin{bmatrix} \text{cat} & \text{ap} \\ \text{lex} & \text{“HaSmaly”} \end{bmatrix} \\ \text{size} & \begin{bmatrix} \text{lex} & \text{“gadol”} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

In this example, *makdeHa HaSmalyt gadola* is produced instead of * *makdeha gadola ve-HaSmalyt* because the *electric* modifier realizes the smixut-licensing relation of `instrument`. This phenomenon gives a further justification for the use of semantic relations in the input.

4.2 Classification of Semantic Relations

Since the syntactic realization component can make good use of a semantic classification in the input, we have designed the classification shown in Table 1, which synthesizes the lists provided by Levi, Glinert and Azar.

In the table we present a basic list of relations with its occurrence percentage in our corpus. It can be viewed that some relations are much more productive than others - purpose, has-part. Our classification is finer than Levi's in distinguishing for example among different types of typical possessive relations (human-relator, has-part and ownership). This reflects slight differences in the default way of generation. Human-relator, for instance, is used as construct when the modifier is a pronoun, more often than with other ownership relations.

[3] We discuss below the heuristics HUGG uses to decide between these 2 paraphrases.

Relation	%corpus	Example
nominalization	12.00%	hacHaSat meZi'ut / denial reality / reality denial
purpose	11.24%	Simlat Hatuna / dress wedding / wedding dress
has-part	11.24%	weyney ha-yeled / eyes the-boy / the boy's eyes
location	6.50%	pirHey midbar / flowers desert / desert flowers
content	6.21%	wugat tapuHym / cake apples / apple cake
human-relator	6.21%	Em habanym / mother the-sons / mother of the sons
type	5.91%	regeS Ahavah / feeling love / love feeling
owner	5.91%	mytat horay / bed my-parents / my parents' bed
producer	5.02%	reyaH bSamym / scent perfium / perfume scent
matter	4.14%	miSpat reZaH / trial murder / murder trial
material	3.84%	cise weZ / chair wood / wooden chair
idioms	3.84%	cadur ha-AreZ / ball the-land / earth
relational	3.55%	Zevaw ha-baZek / color the-batter / the color of the batter
name	2.95%	miSpaHat netanyahu / family netanyahu / The Netanyahu family
experiencer	2.95%	ce'ev-a / pain-her-acc/ her pain
config-units	1.45%	zer praHym / bouquet flowers / a bouquet of flowers
represented	1.18%	semel yokrah / symbol prestige / prestige symbol
part-of	0.88%	nawaley wakev / shoe heel / high-heels shoes
time	0.88%	AruHat Zaharym / meal noon / dinner
cause	0.88%	macat HaSmal / hit electricity / electric shock
caused-by	0.29%	yetuS kadaHat / mosquitoe malaria/ malaria mosquitoe
product	0.29%	mifwal keramika / factory ceramics / ceramics factory
instrument	0.29%	magheZ edim / iron steam / steam iron

Table 1: Semantic relations that can produce a smixut

4.3 Validation of the Classification

To validate empirically the definition of our semantic classification, we gathered a corpus of 853 complex NPs (NPs with more than one modifier) from written Hebrew sources (newspaper and novels). For each NP, we labeled the relations head-modifier in terms of the relations listed in Table 1.

Our evaluation covers two aspects: we first verify that human coders agree on the labeling; we then verify that HUGG can generate from a labeled input a realization similar to that observed in the corpus.

Preliminary evaluation of the agreement among human judges shows agreement of about 90% between three judges (we are currently extending the number of judges). The percentage agreement includes a category "undecided" which covers about 5% of the cases. This corresponds to cases where judges found the relation ambiguous or unclear. Judges agreed on the labelling of unclear relations.

In our corpus, we observed the following distribution in terms of syntactic realization (this takes into account NPs with more than one modifier, explaining that the sum is > 100%):

39%	smixut	31%	pp-qualifier
34%	describer	8%	relative clause

When regenerating from the labeled input we have determined, HUGG's decision to generate a smixut corresponded to that observed in the corpus on more than 95% of the cases.

5 Limitations of a Semantic Account

While the semantic account described above provides good results, it cannot be the only mechanism licensing the production of smixut. We discuss in this section the type of interaction that must be allowed between discourse and pragmatic parameters and the syntactic realization component.

In [6], the interaction between lexical semantics and prgmatics is explored, and two axioms are proposed to interface between the defaults of the lexical semantic and the arbitary knowledge of pragmatics: (1) defaults survive and (2) discourse wins. A statistical method is then added in order to resolve possible interpretations. It is assumed, then, that the grammar/lexicon delimits the range of compounds and indicates conventional interpretations, but that some compounds may only be resolved by pragmatics and that non-conventional contextual interpretations are always available. To provide interpretations, a general schema is encoded in the lexicon leaving undecidable cases to be resolved by pragmatics. Probablities of possible interpretations are taken from corpus frequencies. Accordingly, a new rule is added: (3) Prefer Frequent Senses, which can still be overriden by contextual factors.

From the generation perspective, the interaction between discourse licensed-relations and conventional readings must similarly be controlled by preference rules.

For example, when refering to a city destroyed by the Barbarian, discourse readings cannot override the conventional reading in *The Barbarian city*: discourse cannot force the reading of *a city destroyed by the Barbarians*.

This indicates that a non-monotonic form of reasoning, taking into account preference rules similar to that identified in [6] must be implemented at the pragmatic level. This strongly indicates that this type of reasoning does not belong within the syntactic realization component. Therefore, we conclude that the feature `use-smixut` remains a necessary part of the input specification language to the syntactic realization component.

6 Conclusion

We have presented in this paper basic data on the Hebrew smixut construction. Our strategy to implement smixut in the HUGG syntactic realization is to provide a simple semantic classification in the input. We have demonstrated the many benefits this classification has within the realization process.

Two main problems have been traditionally associated with such semantic accounts of noun-compounding: the relations are not well-defined enough and they are not necessary nor sufficient to explain all uses of compouding. We address these two problems in three ways: (1) we provide an empirical evaluation demonstrating high coder agreement when labeling complex NPs with the set of relations we identify; (2) we demonstrate empirically that the default strategy of "generating a smixut when a semantic relation licenses it" corresponds with the observed usage of smixut in more than 95% of the cases; and (3) we allow the pragmatic module to add a feature `use-smixut` in the input.

The same set of semantic relations is now being used in an extension to SURGE to allow similar paraphrasing decisions in English.

References

[1] Talila Atias. The order of adjectives in hebrew. Master's thesis, Tel Aviv University, Israel, 1981. (in Hebrew).

[2] M. Azar. Classification of hebrew compounds. In R. Nir, editor, *Academic Teachinng of Contemporary Hebrew*. International Center for University Teachinng of Jewish Civilization, Jerusalem, 1985. (in Hebrew).

[3] Ruth Berman. Nominalizations and infinitives in modern hebrew. *Leshunenu*, 39, 1975. (in Hebrew).

[4] Ruth Berman and Dorit Ravid. On the lexical degree of the constructs in hebrew. *Hebrew Computational Linguistics*, (24):5–22, 1986. (In Hebrew).

[5] Hagit Borer. On the morphological parallelism between compounds and constructs. Department of Cognitive Science, School of Social Sciences, University of California, Irvinne.

[6] Ann Copestake and Alex Lascarides. Integrating symbolic and statistical representation: The lexicon pragmatics interface. In *ACL-97*, pages 136–143. ACL, 1997.

[7] P. Downinig. On the creation and use of english compound nouns. *Language*, 53(4):810–842, 1977.

[8] Michael Elhadad. *Using argumentation to control lexical choice: a unification-based implementation*. PhD thesis, Computer Science Department, Columbia University, 1992.

[9] Michael Elhadad. Lexical choice for complex noun phrases: Structure, modifiers and determiners. *Machine Translation*, 11:159–184, 1996.

[10] David Gil. Stacked adjectives and configurationality. *Linguistic Analysis*, (12):141–158, 1983.

[11] L. Glinert. *The Grammar of Modern Hebrew*. Cambridge University, 1989.

[12] J.N. Levi. A semantic analysis of hebrew compound nominals. In Peter Cole, editor, *Studies in Modern Hebrew syntax and semantics*. North-Holland, Amsterdam, 1976.

[13] J.N. Levi. *The Syntax and Semantics of Complex Nominals*. Academic Press, New York, 1978.

[14] I.A. Mel'cuk and N.V. Perstov. *Surface-syntax of English, a formal model in the Meaning Text Theory*. Benjamins, Amsterdam/Philadelphia, 1987.

[15] Uzi Ornan. *The Nominal Phrase in Modern Hebrew*. PhD thesis, Hebrew University, Jerusalem, 1964. (in Hebrew).

[16] Uzi Ornan. *The Simple Clause*. Akademon, Jerusalem, 1979. (in Hebrew).

[17] James D. Pustejovsky. The generative lexicon. *Computational Linguistics*, 17(4):409–441, December 1991.

[18] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. *A comprehensive grammar of the English language*. Longman, 1985.

[19] Channa Seikevicz. *The Possessivve Construction in Modern Hebrew: A Sociolinguistic Approach*. PhD thesis, Georgetown University, Washington D.C., 1979.