LexPar: A Freely Available English Paraphrase Lexicon Automatically Extracted from FrameNet

Bob Coyne
Department of Computer Science
Columbia University
coyne@cs.columbia.edu

Owen Rambow
CCLS
Columbia University
rambow@ccls.columbia.edu

***Abstract*—This is a paper about a new resource, namely an English paraphrase dictionary extracted from the FrameNet lexicon and its example data base.**

I. The LexPar Paraphrase Dictionary

This paper describes LexPar, a lexical resource for paraphrasing English verbs. Paraphrasing within a language can be useful for various applications, such as machine translation (source-side paraphrasing can increase the likelihood of finding a good translation), multi-document summarization (paraphrasing can help find passages in different documents with the same meaning), information extraction (paraphrasing can help in detecting relevant information from seed search patterns), or dialog systems and other generation applications (paraphrasing can make the output more context-appropriate and less monotone). As a result, there has recently been some interest in detecting paraphrases automatically. The main problem here is is the lack of resources: there are few parallel English-English texts (different translations from the same foreign source are one example), and the range of phenomena in paraphrasing is considerable (complex lexico-syntactic paraphrases), so that it can be difficult to generalize.

We take a different approach: we use an existing resource, FrameNet [1], to extract a list of paraphrases for verbs. We associate two verbs if their meanings overlap in a core meaning, even if the mapping from semantic arguments to syntactic arguments is quite different. A typical example is formed by *buy* and *sell*: *X buys Y from Z* and *Z sells Y to X* are paraphrases because they describe the same underlying situation. In order to exploit such a relation, we need not only know the pair ⟨ buy, sell ⟩, but we also need to know the mapping of the syntactic arguments. In FrameNet, verbs that relate to the same underlying semantics are grouped into a frame, with their syntactic arguments mapped to a set of semantic labels specific to the frame. In theory, this should make the extraction of a paraphrase dictionary simple. In practice, there are three important problems: first, *buy* and *sell* are NOT, in fact, in the same frame, but in frames that are related in one of many possible ways; second, verbs in the same frame may not be paraphrases of each other, such as *walk* and *swim*; third, the syntactic annotation in FrameNet (which is needed in order to determine the active-voice valence pattern) is not deep, and not entirely reliable.

LexPar encompasses both the transformations of the conversive lexical function of Meaning-Text Theory (MTT) [2] and the verb alternations modeled in Beth Levin's verb classes [3] and in VerbNet [4]. In addition, it further generalizes these by including mappings from one set of prepositional phrases to others that reference the same underlying semantic roles (*John walked* ***across*** *the field* and *John walked* ***through*** *the field*). We do not include pure synonyms for now which do not involve any syntactic changes (*BUY* and *PURCHASE*), but we easily could if this seems useful. Here is an example.

FROM: ((Buyer Subj) (Goods Obj) (Seller PP/from))
Verb: *BUY.v* **Frame**: COMMERCE_BUY
Example: *Four years ago I* ***bought*** *an old Harmony Sovereign acoustic guitar for 20 pounds from an absolute prat.*
TO: ((Seller Subj) (Goods Obj) (Buyer PP/to))
Verb: *SELL.v* **Frame**: COMMERCE_SELL
Example: *Can't you* ***sell*** *the factory to some other company?*

II. Related Work

Our work falls in the tradition of work on paraphrase generation in the framework of Meaning-Text Theory [5]. This work relies on the existence of an MTT-style lexicon [6], [7], whose creation requires immense human resources. In contrast, we attempt to reuse an existing hand-created resource.

In terms of automatic extraction, [8] present a system that uses syntactic analyses to extract lexico-syntactic paraphrase patterns from corpora. Their approach in some sense subsumes our conversive patterns, since they allow for much more complex transformations that affect more than just two verbs. However, they only allow two arguments (though presumably the work presented could be extended to handle more than two). More importantly, their work is in a different spirit: while we wish to exploit an existing resource as much as possible, they attempt to create a resource entirely automatically from scratch; as a result and as expected, they have a much greater coverage, but at the cost of more noise. Interesting future work could be to investigate how manually created paraphrase resources can be used in conjunction with automatic methods to improve the precision of the latter. Similar comments apply to other work based on parallel (translation) or comparable corpora, such as [9], [10].

Other related work, in a different vein, is that of [11]. Like us, they extract and normalize verb valence patterns by adopting heuristics to determine the voice of FrameNet's unmarked passives. Their larger objective, however, is to derive a syntactic lexicon from FrameNet, whereas our goal is to detect paraphrase pairs. Therefore, our work also differs in that we utilize FrameNet's frame relations in combination with WordNet in order to derive semantic relations between these valence patterns.

III. FrameNet

FrameNet is a digital lexical resource for English that groups related words together into semantic frames. FrameNet currently contains over 10,000 lexical units (nouns, verbs, and adjectives), which correspond roughly to separate lexemes (including fine-grained sense distinctions). Each lexical unit is contained in one of nearly 800 hierarchically-related semantic frames, where each frame represents shared meaning between the lexical units in that frame. In addition, each lexical unit contains a set of annotated sentences which map the sentences' constituent parts to their frame-based roles. FrameNet, in total, contains over 135,000 annotated sentences across all lexical units. Not all lexical units have been annotated. For example, of the approximately 4,100 verbs in FrameNet, only about 2,800 have annotated sentences.

A FrameNet frame consists of a set of frame-based roles, called *frame elements* (FEs). For example, the COMMERCE_SELL frame includes frame elements for SELLER, GOODS, and BUYER. These and other FEs represent the key roles that characterize the meaning of the lexical units in that frame. Frames can contain any number of individual lexical units. The COMMERCE_SELL frame, for example, has lexical units for the words RETAIL, SELL, VEND, etc.

The exact expression of FEs for a given annotated sentence constitutes what FrameNet refers to as a *valence pattern*. In this paper we represent valence patterns as lists of FE and grammatical function (GF) pairs. Grammatical functions are subject (*subj*), object (*obj*), second object (*obj2*), prepositional phrases with strongly governed prepositions (*PP/to*, *PP/on*, *PP/with*, etc.), and clausal complements with specific syntactic properties (*Dep/VPto*, *Dep/Sfin*, etc.). (The FrameNet syntactic annotation, which we are forced to follow, always calls the first NP following the verb the object, even in the double-object construction.) So, for the verb GIVE, the sentence *John gave the book to Mary* has the valence pattern of: ((Donor Subj) (Theme Obj) (Recipient PP/to)). And *John gave Mary the book* has the valence pattern of ((Donor Subj) (Recipient Obj) (Theme Obj2)). Every verb typically has many valence patterns, representing the various ways that verb can be used in sentences.

FrameNet makes a distinction between "core" FEs (those that are unique or characteristic to the meaning of the frame) and "peripheral" frame elements (which do not uniquely characterize a frame). For example, TIME, LOCATION, and MANNER are typically peripheral FEs since they can be instantiated in any appropriate frame. In contrast, in the COMMERCE_BUY frame (which includes the verbs BUY and PURCHASE), the FEs for BUYER and GOODS are core since they are central and conceptually necessary to the meaning of that frame.

FrameNet frames are related to each other by a fixed set of frame relations. These allow us to find semantically related verbs across frames. In addition, since frames can give arbitrary names to their frame elements, frame relations are used to define the mapping between corresponding frame elements in the related frames. Some of the relevant frame relations are:

INHERITANCE represents an is-a relation between two frames. E.g., ESCAPING inherits from DEPARTING.

PERSPECTIVE_ON links frames which represent two different points-of-view of some other neutral frame. For example the frames for verbs BUY and SELL are related as perspectives on the COMMERCIAL_TRANSACTION frame. Similarly, there are three frames for SHOOT corresponding to *shoot the target*, *shooting the gun*, and *shoot the bullet*. These three frames are related via the PERSPECTIVE_ON relation.

INCHOATIVE_OF and CAUSATIVE_OF encode the relationship between stative frames and corresponding inchoative and causative frames For example, the verb COOL is represented by separate lexical units in the frames for CAUSE_TEMPERATURE_CHANGE (as in *John cooled the apple*) and INCHOATIVE_CHANGE_OF_TEMPERATURE (as in *The apple cooled quickly*).

SUBFRAME: Some frames refer to sequences of other frames. These subframes are related to the parent frame via the SUBFRAME relation. For example, the frame CAUSE_IMPACT contains the lexical unit SLAM (as in *John slammed the car door*). This frame has a subframe IMPACT which contains a separate lexical unit for SLAM (as in *the door slammed shut*).

USING is used in cases in which a part of the child's meaning refers to the parent frame. For example, the COMMUNICATION_NOISE frame is used by verbs where communication takes place via a sound (e.g. the verb CLUCK in *"Sorry, Jimmy," the teacher clucks sympathetically*). To represent this dependency, it is related to the MAKE_NOISE frame via the **USING** frame relation.

Note that verbs in the same frame and related frames can vary significantly in meaning. For example, the SELF_MOTION frame contains a large number of verbs related only by the fact that the SELF_MOVER moves under its own power in a directed fashion without a vehicle. As a result, this frame contains strongly related verbs such as WALK and STROLL but also verbs with very different manner of motion such as SWIM and SWING.

By way of an example, we show how various frames related to the commercial transaction meaning are related in Figure 1.

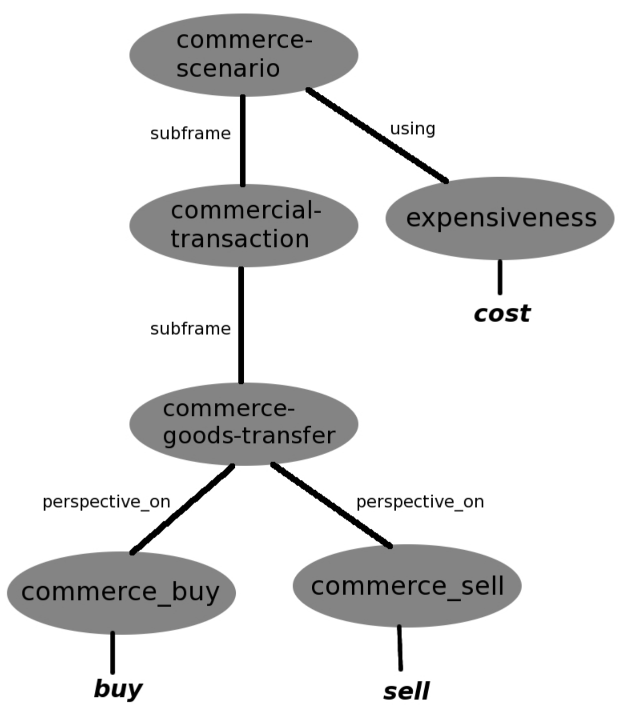

Figure 1. Relation between frames relating to commercial transactions

IV. From FrameNet to LexPar

To generate paraphrase transformations for a given verb we first determine its FrameNet frame and then find related frames using frame relations (see Section III). We then collect a list of potential target verbs in those related frames. Note that the target verb can be the same as the source verb. This allows us produce paraphrases involving diathesis alternations and argument omission.

In selecting potential verbs we rely upon WordNet to test for different types of synonymy. WordNet [12] is a lexical database for the English language. It groups nouns, verbs, and adjectives into sets of synonyms called synsets and arranges those synsets into a hypernym/hyponym hierarchy. So, for example, the synonyms SLEEP and SLUMBER are in the same synset. They are hyponyms of NAP and hypernyms of REST. Note that WordNet synsets and FrameNet lexical units for a given lexeme don't usually correspond exactly. For example, WordNet has a single synset for the verb CLANG, while FrameNet has five lexical units representing separate frames for CAUSE_TO_MAKE_NOISE, CAUSE_IMPACT, IMPACT, MAKE_NOISE, and MOTION_NOISE.

In order to ensure that verb pairs are extended conversives, target verbs must satisfy one of the following conditions:

- The source and target verbs are in the same WordNet synset and hence treated as synonymous. This allows verbs such as HALT and STOP to be paired.
- The source and target verbs are respectively WordNet hyponyms or hypernyms of each other. So, for example, GO is a hypernym of WALK. Thus, sentences like *John walked to the store* and *John went to the store* could be used to describe the same event.
- The source and target verbs are related via the **PERSPECTIVE_ON** frame relation. This covers the case where the two verbs have different surface meaning (and are not related in WordNet) but denote the same underlying event from different points of view.

In this strategy, we will miss some valid paraphrase pairs because of the granularity of meaning in WordNet. For example, in WordNet, BATHE and WASH are not synonyms, hypernyms, or hyponyms of each other. Instead they are both children of CLEANSE which also has unrelated children such FLOSS. As a result we would require some other technique or resource to identify close siblings such as BATHE and WASH in order to include them in our paraphrase pairs.

After collecting the possible target verbs, we then identify all valence patterns for the given source verb. These represent the possible left-hand sides of the transformations. Then, for each of these left-hand side patterns, we collect the right-hand sides from target verb list which have compatible valence patterns. Since we are primarily concerned with paraphrase, the right-hand side valence patterns must only reference FEs that are explicitly expressed in the given left-hand side.

For example, the verb GIVE has the following left-hand side patterns (each pattern consisting of FE and GF pairs):

Possible left-hand side patterns
((Donor Subj) (Recipient Obj) (Theme Obj2))
((Donor Subj) (Theme Obj) (Recipient PP/to))
((Donor Subj) (Theme PP/of) (Recipient PP/to))
((Donor Subj) (Recipient PP/to))

Verbs in related frames have the following valence patterns. As a result, the possible right-hand side patterns will be drawn from the following list:

Possible right-hand side patterns
((Donor Subj) (Theme Obj) (Recipient PP/to))
((Donor Subj) (Theme Obj))
((Donor Subj) (Recipient Obj) (Theme Obj2))
((Theme Subj))
((Theme Subj) (Recipient PP/to))
((Donor Subj) (Recipient Obj) (Theme PP/to))
((Donor Subj) (Theme Obj) (Recipient PP/on))
((Donor Subj) (Recipient Obj))
((Donor Subj) (Recipient Obj) (Theme PP/with))
((Donor Subj) (Theme Obj) (Recipient PP/for))
((Donor Subj) (Theme Obj) (Recipient Obj))
((Donor Subj) (Recipient PP/to))
((Donor Subj) (Theme PP/of) (Recipient PP/to))
((Recipient Subj) (Theme Obj))
((Donor Subj) (Recipient Obj) (Theme Dep/VPto))
((Donor Subj) (Theme Obj) (Recipient PP/upon))

For each possible valence pattern on the left-hand side we collect all valence patterns for the right-hand side that contain only FEs present in the given left-hand side valence pattern. For each of these target valence patterns we list all corresponding verbs (including their lexical unit ID and their frame) along with a matching annotated sentence.

Constructing the valence pattern from the annotated examples in each frame is straightforward, as both the grammatical function and the FE are marked, except for one very important aspect: grammatical voice. The active/passive alternation can be seen as an entirely productive verb alternation in English, and it would make no sense to suggest that every valence pattern for every verb has two additional variants (the passive, and the passive with *by*-agent). Instead, we want to normalize for voice, i.e., we want to always represent the valence pattern for active voice. This is non-trivial, because the syntactic annotation of FrameNet does not include a feature for voice, and the provided grammatical function annotation is for the surface grammatical function. We have implemented a series of heuristics that exploits the part-of-speech and grammatical function annotations, as well as the annotation for missing arguments in passives without *by* agents. However, some cases are impossible to disambiguate for a variety of reasons, including a fair number of examples in which the main verb form is not disambiguated between past tense and past participle and there is no auxiliary (reduced passive relative clause or conjunctions). Thus, grammatical voice is the major source of errors for us in determining the valence pattern.

In constructing these valence patterns we only consider core FEs since these will be characteristic of the verbs in question. Also, in collecting the valence patterns in the mappings from one frame to another, the FE names will often be different. For example, the COMMERCE_SELL frame (used by the verb SELL) has a FE called SELLER, but in the Expensiveness frame (used by the verb COST as in *the book costs 10 dollars*) this identical role is called PAYER. FrameNet's frame relations specify how these different FE names get mapped into each other. We use this information to automatically normalize the names to the namespace of the parent frame. It is the normalized FE names that are output in the paraphrase transformation patterns.

V. RESULTING RESOURCE

Our resulting resource, LexPar, is available online at www1.ccls.columbia.edu/~rambow/resources/lexpar.tar.bz2. LexPar consists of a file for each lexical unit involved on the left-hand side of a paraphrase rule (2,119 files altogether). Each file contains the transformations to other verbs in semantically related frames.

Of the 2,800 annotated verbs in FrameNet (out of 4,100 total), 2,279 are represented in LexPar, on either side of paraphrase transformation rules. The total number of paraphrase transformations among all verbs is 415,479. Of these, 62% retain the same subject FE while 38% assign a different subject FE. 59% of transformations have no object in the source valence pattern, 6% retain the same object FE, and the remaining 25% assign a different object FE. Many paraphrase transformations involve only variations with what preposition is used to refer to a given FE.

Each LexPar file contains of a set of paraphrase instances, each containing one or more paraphrase transformations. Each paraphrase instance represents a unique mapping from a single valence pattern used by the given lexical unit to a different valence pattern used by semantically similar lexical units. The paraphrase instance lists the actual lexical units which match the target valence pattern as well as the frame relation and superframe that were used to derive that valence pattern mapping. When there is a null frame relation, then both the source and target lexical units are in the same frame, and there is no superframe. A threshold level for each lexical unit is also listed. This threshold represents the ratio of the number of times the FE of the subject in the given valence pattern occurs as the subject among all the valence patterns for the given lexical unit. If a FE is only rarely the subject, there is a good chance that it is only playing that role because of a non-detected passive. A trade-off can thus be made between recall and precision by excluding lexical units with a low threshold for the given valence pattern.

For example, the following is a paraphrase instance from a lexical unit for the verb SELL using the **perspective_on** frame relation and mapping the valence pattern ((Seller Subj) (Goods Obj) (Buyer PP/to)) to the valence pattern ((Buyer Subj) (Goods Obj)). The target lexical unit verbs are BUY and PURCHASE. The Threshold entry indicates that SELLER is the subject in 75% of all valence patterns for SELL and that BUYER is the subject 96% of the time for BUY and 92% of the time for PURCHASE.

Valencies: ((Seller Subj) (Goods Obj) (Buyer PP/to)) → ((Buyer Subj) (Goods Obj))
Lexemes: [LU2986 ”sell”] → ([LU2966 ”buy”] [LU2971 ”purchase”])
Thresholds: 0.75 → (0.96 0.92)
Frames: Commerce_sell → Commerce_buy
Frame_relation: PERSPECTIVE_ON
Superframe: Commerce_goods-transfer

So we see that the FE of the subject of X (Y) is the same as the object of Z; this fact, which is made explicit in the rule, allows us to use the rule in applications such as paraphrasing.

After each paraphrase instance, the LexPar file also lists the FrameNet sentences associated with each lexical unit for that valence pattern. The sentences are annotated for voice (active, passive, or passive with a *by*- agent). This is shown in abbreviated form below.

From: [LU2986 ”sell” Commerce_sell] (Threshold: 0.75)
Valence: ((Seller Subj) (Goods Obj) (Buyer PP/to))
(Active) *During the later part of the nineteenth century , the landowners sold the land to developers in very small lots.*
(Active) *And then a woman who had come in to sell flowers to the customers overheard their conversation and intervened.*
To: [LU2966 ”buy” Commerce_buy] (Threshold: 0.96)
Valence: ((Buyer Subj) (Goods Obj))
(Active) *On other occasions , borrowing may be the only way you will ever be able to afford to buy something expensive like a house.*
(Active) *As a result of your win I can buy something special for your ma.*
(Active) *George and Lennie have a dream about their own piece of land which they will be able to buy when they get enough money.*
To: [LU2971 ”purchase” Commerce_buy] (Threshold: 0.92)
Valence: ((Buyer Subj) (Goods Obj))
(Passive-by) *After passing through a number of German private collections it was purchased by the Getty Museum in 1986.*
(Active) *Companies can purchase multiple copies of popular packages at greatly reduced prices.*

We performed an evaluation and error analysis of our results by examining 110 randomly selected transformation rules from our resource. The rules were selected by first randomly selecting a source verb and then randomly selecting a transformation rule for that verb. Once a verb was selected, it was excluded from further selection. Of these 110 rules, 3 were automatically rejected because heuristics found inconsistencies in the syntactic annotation. Of the remaining 107, 79 were judged (by the authors) to be valid. We examine the 28 errors in more detail.

- 17 errors are undetected passive voice constructions, 10 of which were due to wrong POS tags in the FrameNet corpus, and 7 of which are due to errors in our heuristics for passive detection.
- 3 errors are related to the lack of precision in the “CNI” tag used in the FrameNet corpus. It is used to designate syntactically unrealized arguments, and it can occur multiple times; for example, in an embedded passive sentence without *by*-phrase under a control verb, the “CNI” label is used both for the null surface subject and the missing deep subject. This means that we are not able to disambiguate the valence pattern.
- 4 errors are due to wrong annotation of grammatical function in FrameNet.
- 1 error is due to an “easy-to-please” construction, in which our heuristics did not correctly determine the correct deep grammatical function.
- In 2 errors, we think that the semantic annotation is not correct, i.e., a wrong FE was assigned in the corpus.
- In 1 error, our heuristics for following relations between frames resulted in relating two verbs which are clearly not paraphrases. The case is *sting* from the EXPERIENCER_OBJ frame, which is related to *burn* from the EMOTION_HEAT frame. The example sentences are *The words stung Li Yuan , but that was their aim* and *I burn for her*. We judged the meanings of these verbs not to be in a paraphrase relation.

We decided to heuristically eliminate examples if the frequency of the subject FE in an example among all subject FEs is less than a threshold. Varying the threshold gives us a recall-precision trade-off which is summarized below. As we can see, we currently achieve an f-measure of up to 0.93.

Threshold	Recall	Precision	F-Measure
0.0	1.0	.738	.849
0.1	.987	.788	.876
0.2	.975	.856	.911
0.3	.962	.894	.927
0.4	.937	.925	.931

VI. USES OF LEXPAR

There are two main ways in which we see LexPar being used: in a bag-of-words approach (useful for applications such as information retrieval or determining sentence similarity for multi-document summarization), and in a syntactic mode (useful for many applications that perform full syntactic parsing today, including certain approaches to machine translation, or “sentence fusion” for multi-document summarization [13]). In a bag-of-words setting, we could simply use our LexPar as a substitution lexicon: if *teem* is related to *swarm*, then in any bag of words that contains *teem*, we can create an alternative bag of words that contains *swarm* instead. LexPar expands the value of a synonymy resource such as WordNet in a natural way. In systems with full syntactic representation, we can use a parser that generates the deep-syntactic representation we require, such as Minipar [14] or MICA [15]. We can then apply our rules to produce other, semantically equivalent representations. If we simply need syntactically and lexically normalized predicate-argument structures, we are done; if we need surface strings, we subsequently generate from this deep-syntactic representation. We will need to solve algorithmic issues (we have not addressed them in this paper), such as integrating paraphrases into an efficient processing environment.

VII. Conclusion and Future Work

In this paper, we have presented LexPar, a syntactic paraphrase dictionary extracted from FrameNet, and we have presented a small manual evaluation of its quality. That study suggests that the FrameNet resource may become even more useful if its syntactic annotations were checked in an automatic or manual manner (or a combination of both). Various strategies for doing so are conceivable. One fairly straightforward extension to LexPar we envision is to annotate each paraphrase rule with its semantics. The semantics of a paraphrase rule can be derived from two sources: the relation between source frame and destination frame in the FrameNet frame hierarchy, and the relation between the two verbs in WordNet. For the former, see [16] for a semantic classification of such relations. For the latter, we use synonymy, and the hyper- and hyperonymy relations from WordNet, and we could annotate the paraphrase rules directly with these relations. In fact, we could also use the antinomy relation, creating anitomic "paraphrases" (which we do not currently do, but which may prove useful for certain purposes). Finally, we have sketched a range of possible applications for LexPar in Section VI. Of course, LexPar needs to be validated by showing that it can provide a performance increase in at least one of the mentioned applications.

Acknowledgments

We would like to thank Julia Hirschberg and Sylvain Kahane for feedback on the ideas presented in this paper, and three anonymous reviewers for their useful comments. This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under contract number HR0011-06-C-0023, and by the National Science Foundation under grant IIS-0904361. Any opinions, findings and conclusions or recommendations expressed in this material are the authors and do not necessarily reflect those of of the funding agencies.

References

[1] C. F. Baker, J. Fillmore, and J. B. Lowe, "The Berkeley FrameNet project," in *COLING-ACL'98*, 1998.

[2] I. A. Mel'čuk, *Dependency Syntax: Theory and Practice*. New York: State University of New York Press, 1988.

[3] B. Levin, *English Verb Classes and Alternations: A Preliminary Investigation*. The University of Chicago Press, 1993.

[4] K. Kipper, A. Korhonen, N. Ryant, and M. Palmer, "A large-scale classification of English verbs," *Language Resources and Evaluation Journal*, vol. 42, no. 1, pp. 21–40, 2008.

[5] L. Iordanskaja, R. Kittredge, and A. Polguère, "Lexical selection and paraphrase in a meaning-text generation model," in *Natural Language Generation in Artifical Intelligence and Computational Linguistics*, C. Paris, W. Swartout, and W. Mann, Eds. Kluwer Academic Publishers, 1991, pp. 293–312.

[6] I. A. Mel'čuk and A. Polguère, "A formal lexicon," *Computational Linguistics*, vol. 13, no. 3–4, pp. 13–54, 1987.

[7] M. Alonso Ramos and A. Tutin, "Classification and description of lexical functions for the analysis of their combination," in *Lexical Functions in Lexicography and Natural Language Processing*, L. Wanner, Ed. Amsterdam and Philadelphia: John Benjamins, 1996, pp. 147–168.

[8] D. Lin and P. Pantel, "Discovery of inference rules for question answering," *Natural Language Engineering*, vol. 7, no. 4, pp. 343–360, 2001.

[9] R. Barzilay and K. McKeown, "Extracting paraphrases from a parallel corpus," in *39th Meeting of the Association for Computational Linguistics (ACL'01)*, 2001.

[10] R. Barzilay and L. Lee, "Learning to paraphrase: An unsupervised approach using multiple-sequence alignment," in *HLT-NAACL03*, 2003.

[11] M. McConville and M. O. Dzikovska, "'deep' grammatical relations for semantic interpretation," in *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, Manchester, UK, 2008.

[12] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Cambridge, MA: MITPress, 1997.

[13] R. Barzilay, "Information fusion for multidocument summarization: Paraphrasing and generation," Ph.D. dissertation, Columbia University, 2003.

[14] D. Lin, "PRINCIPAR—an efficient, broad-coverage, principle-based parser," in *coling94*, Kyoto, Japan, 1994, pp. 482–488.

[15] S. Bangalore, P. Boullier, A. Nasr, O. Rambow, and B. Sagot, "MICA: A probabilistic dependency parser based on tree insertion grammars," in *NAACL HLT 2009 (Short Papers)*, 2009.

[16] B. Coyne and O. Rambow, "Meaning-text-theory and lexical frames," in *Fourth International Conference on Meaning-Text Theory*, Montréal, QC, Canada, 2009.