Noun Generation for Nominalization in Academic Writing

Dariush Saberi, John Lee
Department of Linguistics and Translation
City University of Hong Kong
dsaberi2-c@my.cityu.edu.hk, jsylee@cityu.edu.hk

Abstract

Nominalization is a common technique in academic writing for producing abstract and formal text. Since it often involves paraphrasing a clause with a verb or adjectival phrase into a noun phrase, an important task is to generate the noun to replace the original verb or adjective. Given that a verb or adjective may have multiple nominalized forms with similar meaning, the system needs to be able to automatically select the most appropriate one. We propose an unsupervised algorithm that makes the selection with BERT, a state-of-the-art neural language model. Experimental results show that it significantly outperforms baselines based on word frequencies, word2vec and doc2vec.

1 Introduction

Automatic paraphrasing — re-writing a sentence while preserving its original meaning — has received much interest in the computational linguistics community in recent years. One type of paraphrasing is lexical substitution (McCarthy and Navigli, 2009), which replaces a word or short phrase with another. Paraphrasing can also involve manipulation of the clausal structure of a sentence, with a range of options that has been described as the "cline of metaphoricity" (Halliday and Matthiessen, 2014). Towards one end of this cline, the text offers a "congruent construal of experience", and the sentences tend to be clausally complex but lexically simple (e.g., the complex clause "Because she didn't know the rules, she died"[1]). Towards the other end of the cline, the text exhibits a "metaphorical reconstrual", and the sentences are clausally simpler and lexically denser (e.g., the nominal group "Her death through ignorance of the rules").

Previous studies on automatic manipulation of clausal structure have mostly concentrated on syntactic simplification, typically by splitting a complex sentence into two or more simple sentences (Siddharthan, 2002; Aluísio et al., 2008; Narayan and Gardent, 2014). More recent research has also attempted semi-automatic nominalization (Lee et al., 2018), which aims to paraphrase a complex clause into a simplex clause by transforming verb or adjectival phrases into noun phrases.

Noun generation is a core task in the nominalization pipeline (Table 2). Resources such as NOMLEX (Meyers et al., 1998) and CATVAR (Habash and Dorr, 2003) have greatly facilitated this task by providing lists of related nouns, verbs and adjectives. However, straightforward look-up in these lists does not suffice since a word may have multiple nominalized forms with similar meaning. For example, the verb "dominate" can be transformed into "domination", "dominance", "dominion", as well as the gerund form "dominating". We will henceforth refer to these as the "***noun candidates***". As shown in Table 1, in the context of the clause "The British dominated India", "domination" would be preferred (i.e., "British *domination* of India"); in the context of the clause "older people dominated this neighborhood", "dominance" would be more appropriate (i.e., "The *dominance* of older people in this neighborhood").

The goal of this paper is to evaluate a noun generation algorithm that selects the best noun candidate during nominalization. The approach taken by Lee et al. (2018), which considers noun frequency statistics alone, always selects the same noun regardless of the sentential context. We use instead a neural language model, BERT, for noun generation. Experimental results show that it significantly outperforms baselines based on word

[1] This example and the next are both taken from Halliday and Matthiessen (2014).

Verb-to-noun mapping	Example sentence	Nominalized version
dominate → {dominance, domination, ...}	The British **dominated** India ... Older people **dominated** this neighborhood ...	British **domination** of India ... The **dominance** of older people in this neighborhood ...
move → {motion, move, ...}	They **moved** northward ... The particle **moved** irregularly ...	Their **move** northward ... The irregular **motion** of the particle ...
enter → {entrance, entry, ...}	The clown **entered** the stage ... The immigrants **entered** the country ...	The clown's **entrance** to the stage ... The **entry** of the immigrants into the country ...
measure → {measure, measurement, ...}	Success is **measured** ... Blood pressure is **measured** ...	The **measure** of success ... The **measurement** of blood pressure ...

Table 1: Example verb-to-noun mappings with multiple *noun candidates* (left column), illustrated by sentences with the same verb (middle column) requiring different *target nouns* (right column) in their nominalized version.

frequencies, word2vec and doc2vec.

The rest of the paper is organized as follows. Following a review of previous work (Section 2), we give details on our dataset (Section 3) and outline our approach (Section 4). We then report experimental results (Section 5) and conclude.

2 Previous work

We first discuss the relation between our task and lexical substitution (Section 2.1) and word sense disambiguation (Section 2.2). We then describe an existing nominalization system (Section 2.3), whose noun generation algorithm will serve as our baseline.

2.1 Relation to lexical substitution

Noun generation in nominalization can be considered a specialized kind of lexical substitution. While lexical substitution typically aims for a paraphrase in the same part-of-speech (POS) (e.g., "dominate" → "prevail"), our task by definition involves a change in POS, usually from a verb or adjective to a noun (e.g., "dominate" → "domination"). This difference is reflected in the limited number of verb-noun or adjective-noun entries in open-source paraphrase corpora such as PPDB (Ganitkevitch et al., 2013).

2.2 Relation to word sense disambiguation

Word sense disambiguation (WSD) is relevant to noun generation to the extent that verb senses can guide the choice of noun candidates. For example, "succeed" in the sense of "achieve the desired result" should be paraphrased as "success" ("He succeeded in ..." → "His success in ..."), whereas "succeed" in the sense of "take over a position" would require "succession" ("He succeeded to the throne ..." → "His succession to the throne ...").

WSD is not necessary for noun generation when the verb corresponds to a noun with the same range of meanings. Consider the verb "conclude", which may mean either "to finish" or "to reach agreement". Nominalization requires no WSD since the noun "conclusion" preserves the same semantic ambiguity.

In other cases, our task requires fine-grained WSD, especially when the noun candidates are semantically close. Their differences can be rather nuanced (e.g., "domination" vs. "dominance"), making it challenging for typical WSD models to distinguish.

2.3 Nominalization pipeline

In the first reported tool for semi-automatic nominalization aimed at academic writing (Lee et al., 2018), the system first parses the input clause to detect the potential for nominalization. If its dependency tree exhibits an expected structure (e.g., Table 2(i)), the system proceeds to lexical mapping (Table 2(ii)), which includes transforming the main verb ("entered") to a noun ("entrance"); an adverb ("abruptly") to an adjective ("abrupt"); and the subject ("the clown") to a possessive form ("the clown's" or "of the clown") . Finally, the system generates a sentence by choosing one of the possible surface realizations through heuristics (Table 2(iii)).

The noun generation task in lexical mapping utilizes verb-to-noun and adjective-to-noun mappings, some examples of which are shown in Table 1. The system constructed these mappings on the basis of NOMLEX (Meyers et al., 1998) and

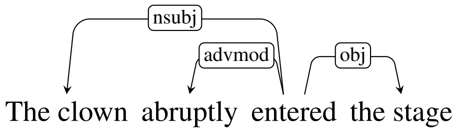
(i) Parsing	 nsubj advmod obj The clown abruptly entered the stage				
(ii) Lexical mapping	the clown ↓ the clown's	abruptly ↓ abrupt	**entered** ↓ **entrance**	the stage ↓ to the stage	
(iii) Sentence generation	The The clown's His	abrupt abrupt abrupt	entrance entrance entrance	of the clown	to the stage ... to the stage ... to the stage ...

Table 2: The nominalization pipeline (Lee et al., 2018): (i) syntactic parsing; (ii) lexical mapping, including noun generation (bolded), which is the focus of this paper; and (iii) sentence generation.

CATVAR (Habash and Dorr, 2003)[2], with a total of 7,879 verb-to-noun mappings, and 11,369 adjective-noun mappings.

3 Dataset

Among the mappings described in Section 2.3, there were 7,380 verb-to-noun and 5,339 adjective-to-noun mappings with at least two noun candidates. We constructed our dataset on the basis of these mappings only, because the others do not require selection from multiple candidates.

The ideal dataset for this research would consist of input sentences containing these verbs and adjectives; and, as gold output, the noun candidate selected for use in the nominalized version of these sentences. Unfortunately, no such large-scale dataset exists. One option is to sample sentences in a corpus and ask human experts to nominalize them; this would however require considerable manual annotation. To avoid this cost, an alternative is to work backwards: identify sentences containing noun phrases that could plausibly be the result of nominalization (e.g., those in the right column of Table 1). This methodology produces the gold noun candidate automatically. One can then retrieve from the mappings the verb or adjective that would be in the hypothetical sentence before nominalization (e.g., those in the middle column of Table 1). Adopting this methodology, we constructed a challenging dataset by prioritizing verbs and adjectives that are more ambiguous, i.e., those with more noun candidates.

One potential issue is the plausibility of the selected sentences as the nominalized form of an input sentence. To make our dataset as realistic as possible, we required sentences to have one of the three common nominalized forms, corresponding to the three surface forms shown in Table 2(iii):

- "the <target noun> of <subject> ..."
- "<subject>'s <target noun> ..."
- "<poss> <target noun> ..."

where <target noun> is the gold noun candidate, <poss> is a possessive pronoun and <subject> is the noun subject of the hypothetical input sentence before nominalization. In addition, we require the target noun, verb and adjective to be tagged as such at least two times in the Brown Corpus (Francis and Kučera, 1979), to avoid words with rare usage.

Our dataset consists of a total of 620 sentences that satisfy the above requirements, including 332 retrieved from the Brown Corpus and 288 from the British Academic Written English (BAWE) Corpus (Nesi, 2008). The sentences contain 73 distinct verbs and 19 distinct adjectives, each with an average of 2.67 noun candidates.

4 Approach

The noun generation algorithm used by Lee et al. (2018) considers only the word frequency statistics of the noun candidates. It therefore always chooses the same noun candidate for a verb (or adjective), even if the sentential context warrants a different choice due to word sense, register or fluency considerations.

To remove this limitation, we use BERT (Devlin et al., 2019), a state-of-the-art neural language model based on the "Transformer" architec-

[2] Verbs-to-be and modal verbs were not treated.

ture (Vaswani et al., 2017). BERT has been shown to be effective in a wide range of natural language processing tasks. The model is bi-directional, i.e., trained to predict the identity of a masked word based on the words both before and after it. We consider the suitability of each noun candidate in the verb-to-noun and adjective-to-noun mappings as the masked word.

In each sentence in our dataset, we mask the target noun and ask BERT for its word predictions for the masked position.[3] Among the noun candidates, we identify the highest-ranked one among the first 15,000 word predictions. If none of the candidates is ranked, we create a sentence with each candidate by replacing the masked word with it, and obtain the BERT score for the sentence. We select the candidate that yields the sentence with the highest score.

5 Results

We compared our proposed approach with four baselines:

Spelling This baseline selects the noun candidate that has the smallest letter edit distance from the original verb or adjective.

Frequency Following Lee et al. (2018), this baseline selects the noun candidate with the highest unigram frequency count in the *Google Web 1T Corpus* (Brants and Franz, 2006).

Word2vec We select the noun candidate that is most similar to the original verb or adjective, as estimated by the Google News pre-trained Gensim model (Mikolov et al., 2013).

Doc2vec We select the noun candidate that has the highest cosine similarity with the sentence embeddings, taking each sentence as a small "document".[4]

As shown in Table 3, the Frequency baseline achieved higher accuracy than the Spelling baseline and Word2vec. The frequency of a noun candidate appears to serve as a good proxy for its appropriateness. All three approaches, however, ignore the specific context of the sentence, always proposing the same noun for a given verb or adjective.

Approach	Brown	BAWE
Frequency	53.92%	48.61%
Spelling	46.39%	35.07%
Word2vec	35.84%	43.71%
Doc2vec	36.74%	38.88%
BERT	**74.10%**	**72.57%**

Table 3: Accuracy of our proposed noun generation algorithm with BERT, compared to baselines.

By taking the rest of the sentence into account when predicting the noun candidate, BERT yielded better performance. Consider the verb "measure". Although frequency favors the noun "measure", BERT was able to select "measurement" when it collocates with "quantity". While Doc2vec also considers the sentential context, it did not perform as well as BERT, likely because the masked language modeling objective offers a better fit for our task.

Still, BERT's performance was limited by difficulties in recognizing nuanced differences between noun pairs such as "use" and "usage", or "occupation" and "occupancy". With access only to a single sentence, it was also unable to choose formal words such as "continuance" over "continuation" when called for by the context.

6 Conclusion

We propose an unsupervised algorithm for noun generation from a verb or adjectival phrase, a task that is essential for automatic nominalization system for academic writing. This algorithm selects the most appropriate noun candidate with BERT, a state-of-the-art neural language model. Experimental results show that it significantly outperforms baselines based on word frequencies, word2vec and doc2vec.

Acknowledgments

This work was partially funded by a HKSAR UGC Teaching Learning Grant (Meeting the Challenge of Teaching and Learning Language in the University: Enhancing Linguistic Competence and Performance in English and Chinese) in the 2016-19 Triennium; and by an Applied Research Grant (#9667151) from City University of Hong Kong.

[3] We used the PyTorch implementation of BERT with the bert-base-uncased model.

[4] We used the following settings: max epocs = 100, vector size = 20, alpha = 0.025, min count = 1, dm = 1. With word embeddings combined, the best results were obtained with dbow = 0 and dmpv = 0

References

Sandra Aluísio, Lucia Specia, T. A. Pardo, E. G. Maziero, and R. P. Fortes. 2008. Towards Brazilian Portuguese Automatic Text Simplification Systems. In *Proc. 8th ACM Symposium on Document Engineering*.

Thorsten Brants and Alex Franz. 2006. The Google Web 1T 5-gram Corpus Version 1.1. In *LDC2006T13*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding. In *Proc. NAACL-HLT*.

W. N. Francis and H. Kučera. 1979. Manual of Information to Accompany a Standard Corpus of Present-Day Edited American English, for use with Digital Computers. Providence, RI. Department of Linguistics, Brown University.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *Proc. NAACL-HLT*.

Nizar Habash and Bonnie Dorr. 2003. A Categorial Variation Database for English. In *Proc. NAACL*.

M. A. K. Halliday and C. M. I. M. Matthiessen. 2014. *Halliday's Introduction to Functional Grammar*. Routledge.

John Lee, Dariush Saberi, Marvin Lam, and Jonathan Webster. 2018. Assisted Nominalization for Academic English Writing. In *Proc. Workshop on Intelligent Interactive Systems and Language Generation (2ISNLG)*, pages 26–30.

Diana McCarthy and Roberto Navigli. 2009. The English Lexical Substitution Task. *Language Resources and Evaluation*, 43:139–159.

Adam Meyers, Catherine Macleod, Roman Yangarber, Ralph Grishman, Leslie Barrett, and Ruth Reeves. 1998. Using NOMLEX to Produce Nominalization Patterns for Information Extraction. In *Proc. Computational Treatment of Nominals*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Proc. International Conference on Learning Representations (ICLR)*.

Shashi Narayan and Claire Gardent. 2014. Hybrid Simplification using Deep Semantics and Machine Translation. In *Proc. ACL*.

Hilary Nesi. 2008. BAWE: an introduction to a new resource. In *Proc. Eighth Teaching and Language Corpora Conference*, page 239–46, Lisbon, Portugal. ISLA.

Advaith Siddharthan. 2002. An Architecture for a Text Simplification System. In *Proc. Language Engineering Conference (LEC)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. *Advances in Neural Information Processing Systems*, pages 6000–6010.