



香港城市大學
City University of Hong Kong

專業 創新 胸懷全球
Professional · Creative
For The World

CityU Scholars

Automatic Nominalization of Clauses through Textual Entailment

Lee, John S. Y.; Lim, Ho Hung; Webster, Carol; Melser, Anton

Published in:

Proceedings of the 29th International Conference on Computational Linguistics

Published: 01/10/2022

Document Version:

Final Published version, also known as Publisher's PDF, Publisher's Final version or Version of Record

License:

CC BY

Publication record in CityU Scholars:

[Go to record](#)

Publication details:

Lee, J. S. Y., Lim, H. H., Webster, C., & Melser, A. (2022). Automatic Nominalization of Clauses through Textual Entailment. In *Proceedings of the 29th International Conference on Computational Linguistics* (pp. 6002-6006). International Committee on Computational Linguistics. <https://aclanthology.org/2022.coling-1.524/>

Citing this paper

Please note that where the full-text provided on CityU Scholars is the Post-print version (also known as Accepted Author Manuscript, Peer-reviewed or Author Final version), it may differ from the Final Published version. When citing, ensure that you check and use the publisher's definitive version for pagination and other details.

General rights

Copyright for the publications made accessible via the CityU Scholars portal is retained by the author(s) and/or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights. Users may not further distribute the material or use it for any profit-making activity or commercial gain.

Publisher permission

Permission for previously published items are in accordance with publisher's copyright policies sourced from the SHERPA RoMEO database. Links to full text versions (either Published or Post-print) are only available if corresponding publishers allow open access.

Take down policy

Contact lbscholars@cityu.edu.hk if you believe that this document breaches copyright and provide us with details. We will remove access to the work immediately and investigate your claim.

Automatic Nominalization of Clauses through Textual Entailment

John S. Y. Lee¹, Ho Hung Lim¹, Carol Webster², Anton Melser¹

¹ Department of Linguistics and Translation, City University of Hong Kong

² College of Professional and Continuing Education, The Hong Kong Polytechnic University
Hong Kong SAR, China

{jsylee,hhlim3}@cityu.edu.hk, carol.webster@cpce-polyu.edu.hk,
anton.melser@my.cityu.edu.hk

Abstract

Nominalization re-writes a clause as a noun phrase. It requires the transformation of the head verb of the clause into a deverbal noun, and the verb’s modifiers into nominal modifiers. Past research has focused on the selection of deverbal nouns, but has paid less attention to the word order and word forms for the nominal modifiers. We propose using a textual entailment model for clause nominalization. Experimental results show that a textual entailment model fine-tuned on this task outperforms a number of unsupervised approaches using language model scores.

1 Introduction

Many textbooks on academic writing devote significant attention to nominalization, which lends a more abstract, concise and objective tone to a text (Kamler and Thomson, 2006; Bailey, 2011). Since nominalization requires careful lexical selection and clause restructuring, it demands advanced vocabulary knowledge and grammatical skills, making it challenging even for many human writers. Table 1 outlines the steps in nominalizing a sentence: extraction of target clauses; nominalization of a target clause through word re-ordering and re-generating the POS and prepositions; and the re-writing of the sentence to incorporate the nominalized clauses. This paper focuses on the clause nominalization step.

Past research on clause nominalization has concentrated on replacement of the head verb with a deverbal noun (e.g. ‘omits’ → ‘omission’) and resource development to support the task (Meyers et al., 1998; Habash and Dorr, 2003; Saberi and Lee, 2019). Less attention has been paid to the clause restructuring that is required for transforming the verb’s modifiers in the clause to nominal arguments, including word reordering (e.g., postposing the subject ‘Arabic’, Table 1i); POS re-generation of the verb’s modifiers (‘frequently’ →

Extract clause	Since [<i>Arabic frequently omits vowels</i>], the surname is often spelt ...
Nominalize clause	(i) Word-order edits: [frequently vowels omits Arabic]
	(ii) POS Re-generation: [frequent vowels omission Arabic]
	(iii) Preposition generation: [frequent vowel omissions in Arabic]
Re-write sentence	Due to [frequent vowel omissions in Arabic], the surname is often spelt ...

Table 1: Main steps in nominalizing a sentence, focusing on the target clause enclosed in [...].

‘frequent’, Table 1ii); and preposition generation (Table 1iii).

This paper investigates methods for clause nominalization that optimize the position and POS of the nominal arguments. We focus on clauses headed by a verb with up to three syntactic arguments, including subjects, objects, adjectival phrases or prepositional phrases. The output is a nominalized form of the clause that preserves its original meaning. Experimental results show that a textual entailment model that is fine-tuned on this task can outperform unsupervised approaches based on neural language model scores.

The rest of the paper is organized as follows. After a review of previous work (Section 2), we present our two-step algorithm: candidate generation (Section 3) followed by candidate ranking (Section 4). Finally, we report experimental results (Section 6).

2 Previous Work

Shinyama et al. (2002) explored automatic acquisition of paraphrase templates, which included nominalizations. Lee et al. (2018) evaluated the template-based approach specifically on nominalization, but relied on heuristics for word order, POS and preposition selection. Fujita and Sato (2008)

Edit	Rewrite	Example input	Gold nominalization
$\neg V$	$V \rightarrow \emptyset$	(a) she ^S makes ^V a decision ^{O₁} (b) Aunt Lella ^S fell ^V ill ^{O₁}	her decision Aunt Lella’s illness
$S \Rightarrow$	$SVO_1 \rightarrow VO_1S$ $SV \rightarrow VS$	(c) citizens ^S initiate ^V plebiscites ^{O₁} (d) Mexico ^S suffers ^V high casualties ^{O₁} (e) results ^S were ^V inconclusive ^{O₁} (f) tension ^S increases ^V in the region ^{O₁}	initiation of plebiscites by citizens high casualties for Mexico inconclusive results an increase in tension in the region
$O_1 \Leftarrow$	$VO_1 \rightarrow O_1V$	(g) transfer ^V money ^{O₁} to the hijackers ^{O₂} (h) Mubarak ^S raises ^V taxes ^{O₁}	money transfer to the hijackers Mubarak’s tax raise
$O_2 \Leftarrow$	$VO_1O_2 \rightarrow O_2VO_1$	(i) use ^V animals ^{O₁} for research ^{O₂} (j) use ^V the park ^{O₁} for recreation ^{O₂}	research use of animals recreational use of the park

Table 2: Word-order edits on example target clauses towards the generation of the gold nominalization

automatically generated syntactic variants of predicate phrases, using n-gram models and distributional similarity measures to estimate semantic equivalence and syntactic substitutability.

Our task is distinct from semantic role labeling (Lapata, 2002; Padó et al., 2008), since it must take into account both the fluency of the nominalization and its semantic equivalence to the original clause. Another related task, the paraphrasing of nominalizations, can be viewed as the reverse of ours. In an unsupervised approach, Lee et al. (2021) selected the best clausal paraphrase of a nominalization using a language model and a textual entailment model, which has also been applied to other NLP tasks such as question answering (Trivedi et al., 2019), summarization (Kryściński et al., 2020), and relation extraction (Sainz et al., 2021). However, the candidate generation algorithm for our task is significantly different. We also address a wider range of nominalization patterns in our evaluation, and show that a supervised approach can improve performance.

3 Nominalization Generation

As shown in Table 2, the input is a sentence with a target clause consisting of the head verb (V) with up to three syntactic arguments, which may be its subject (S) or other arguments in object position (O₁, O₂).¹ Candidate nominalizations are generated with the following steps:

3.1 Word-order edits

Table 2 lists all edit operations for re-positioning S and O_i in relation to the verb V, including:

¹The clauses would need to be automatically extracted in actual deployment. We assume accurate extraction of these strings to avoid confounding the experimental results with parsing errors.

Delete verb ($\neg V$) Nominalizations may omit the support verb or light verb in the clause (Table 2a). The copula or a semantically blanched verb can also be omitted, typically when O₁ is an adjective (Table 2b).

Postpose subject ($S \Rightarrow$) The subject can be positioned after the verb to serve as a postnominal modifier of the deverbal noun (Table 2c), or of O₁ when V is deleted (Table 2d). It may also head the nominalized clause, typically when O₁ is an adjective and V is deleted (Table 2e).

Prepose object ($O_i \Leftarrow$) The O₁ (Table 2g-h) or O₂ (Table 2i-j) can be moved in front of V to become a prenominal modifier of the deverbal noun.

3.2 POS Re-generation

Head noun generation. The verb V can be substituted with the deverbal noun (e.g. ‘omits’ → ‘omission’ in Table 1ii) to head the nominalized clause. If V is deleted, an O₁ adjective can be substituted with a deadjectival noun (e.g., ‘ill’ → ‘illness’ in Table 2b). All nouns that are derivationally related to the verb or adjective in WordNet (Fellbaum, 2010) are considered candidates.

Prenominal modifier generation. The most fluent nominalization may require different POS for the prenominal modifier depending on context. It may be an adjectival form of the S (Table 3a), of the preposed O_i (Table 3b), or of the adverb (Table 3c). All adjectives listed in WordNet as its pertainyms or derivationally related forms are considered candidates. The prenominal modifier may also be an s-genitive (Table 3d), a possessive pronoun (Table 3e), or the singular form of the original noun (e.g. ‘vowels’ → ‘vowel’ in Table 1ii).

Rewrite	Example input	Gold nominalization
Adjective	(a) Americans immigrate (b) control the diet of the patient (c) the member leaves suddenly	American immigration dietary control of the patient the sudden departure of the member
Genitive	(d) the city emerges from default	the city’s emergence from default
Poss. Pron.	(e) it experiences a mild climate	its mild climate

Table 3: Part-of-speech (POS) edits that re-write a noun in the input as an adjective, s-genitive or possessive pronoun to serve as pronominal modifier in the nominalization

3.3 Preposition and determiner generation

Prepositions and determiners can be inserted in front of postmodifiers (Table 1iii), and a determiner may be inserted at the front of the nominalized clause. We enumerate the permutations of all choices of determiners (‘a’, ‘an’, ‘the’), and use the masked language model BERT (Devlin et al., 2019) to predict the most likely preposition.²

4 Candidate ranking

A pool of candidate nominalizations are generated for each target clause using all permutations described in Section 3. We evaluated the following methods to select the best candidate from the pool:

Language Model (LM) Select the candidate that yields the highest-scoring sentence, according to the log-probability score based on GPT-2 (Salazar et al., 2020).³

Majority Same as the above but consider only those candidates of the majority pattern, i.e., retain the original word order SVO_1O_2 and use the s-genitive as the pronominal modifier.

Pre-trained TE Model The premise is the input sentence, and the hypothesis is the sentence re-written using the candidate nominalization (Table 1). The TE model predicts whether the premise implies the hypothesis. We used AllenNLP’s pre-trained TE model on SNLI based on RoBERTa (Liu et al., 2019).⁴ Similar to the approach proposed by Lee et al. (2021),

we identify the two candidates with the highest TE model score, and select the one with the higher LM score.

Fine-tuned TE Model Same as above, except that the pre-trained model was fine-tuned on our dataset (Section 5).⁵ For each premise, the gold outputs served as the ‘entailment’ hypotheses. There were on average 252 candidate nominalizations in the candidate pool. For each non-gold word order (Section 3), we selected the candidate with the highest LM score to serve as a ‘neutral’ hypothesis. There were on average 7.8 ‘neutral’ hypotheses for each premise.

5 Data

Since our research focus is at the clause level rather than the re-writing of the entire sentence, we targeted sentences that permit straightforward alternation between a clause and a noun phrase through: (1) a change of conjunction, e.g., “although ⟨clause⟩” ↔ “despite ⟨NP⟩”; (2) a verb that can take both clause or noun phrase as argument, e.g., “report that ⟨clause⟩” ↔ “report ⟨NP⟩”; and (3) a clause that can replace a discourse deixis as a noun phrase, e.g., “⟨clause⟩. That ⟨verb⟩ ... ” ↔ “⟨NP⟩ ⟨verb⟩ ...”.

Our dataset contains 319 unique inputs and 751 nominalizations (Table 4).⁶ Among the inputs, 202 were extracted from Wikipedia (Section 5.1) and 117 from an existing dataset (Section 5.2).

5.1 Nominalization annotation

We retrieved sentences with the above three patterns from Wikipedia. For efficient annotation, we collected sentences with at least one clause headed by a verb with a derivationally related noun in WordNet. Four annotators, all university students

²Downloaded from <https://storage.googleapis.com/allennlp-public-models/bert-masked-lm-2020-10-07.tar.gz>

³We used the 345M version of GPT-2 from <https://github.com/awsmlabs/mlm-scoring>

⁴Downloaded from <https://storage.googleapis.com/allennlp-public-models/snli-roberta.2021-03-11.tar.gz>. We also tried pre-trained TE models based on DeBERTa (<https://huggingface.co/microsoft/deberta-v2-xlarge-mnli>) and those provided by Nie et al. (2020) but achieved lower accuracy than our best model.

⁵We used default values for all parameters during fine-tuning.

⁶Accessible at <https://github.com/NominalizationParaphrase>

Initial word in gold nominalization	Pre. modifier			Total
	N	G	A	
Nominalized verb	n/a	n/a	n/a	221
Subject (S)	70	140	92	302
Object (O ₁ or O ₂)	123	18	87	228

Table 4: Distribution of different word orders and POS of the prenominal modifier, as noun (N), adjective (A), s-genitive or possessive pronoun (G) in the gold nominalizations in our dataset

who were native speakers of English, composed possible nominalizations, if any, for each sentence. The annotators were asked to favor derivationally related forms over free paraphrases. Only those nominalizations produced by at least two annotators were included in the dataset.

5.2 Conversion from paraphrase dataset

The nominalization-clause pairs in the dataset produced by Lee et al. (2021) are not directly usable, since the clause paraphrases only the nominalization but not the rest of the sentence. We identified the sentences in this dataset with the three patterns described above, and then replaced the nominalization with their gold clause according to the alternation templates above to produce an input sentence for our task.⁷

Since the nominalization inputs in this dataset all have one prenominal modifier and one prepositional phrase, they would lead to an imbalanced output in our dataset. We asked a native speaker of English, a PhD candidate in linguistics, to enumerate other acceptable nominalizations, which were then reviewed by a professor of linguistics who was a near-native speaker of English.

6 Experiments

6.1 Set-up

All models were evaluated on the full dataset (Section 5), with 10-fold cross-validation used for the Fine-tuned TE Model. In the **Gold** setting, the gold forms of the prenominal modifier and deverbal noun were always included as one of the candidates along with other candidates retrieved from WordNet. The **Auto** setting was fully automatic.

We report three evaluation metrics in ascending strictness. For **word order accuracy**, the output

⁷e.g., the sentence “Due to frequent vowel omissions in Arabic ...” is re-written as “Since Arabic frequently omits vowels ...” to serve as the input in our dataset.

Setting	Metric → ↓ Model	WO acc.	POS acc.	Nom acc.
Gold	Majority	0.467	0.429	0.376
	LM	0.586	0.536	0.476
	Pre-trained TE	0.639	0.524	0.439
	Fine-tuned TE	0.812	0.743	0.630
Auto	Majority	0.429	0.395	0.323
	LM	0.514	0.476	0.389
	Pre-trained TE	0.586	0.470	0.326
	Fine-tuned TE	0.724	0.655	0.511

Table 5: Model performance in the Gold and Auto settings in terms of word order (WO) accuracy, POS accuracy and nominalization accuracy

is considered correct if it matches the gold word order (Section 3.1). **POS accuracy**, in addition, requires all prenominal modifiers to have the gold POS (Section 3.2). **Nominalization accuracy** requires the lemmatized form of the output to match exactly with the gold, except determiners.

6.2 Results

As expected, system performance was higher in the Gold setting than Auto (Table 5). In both settings, the use of LM scores yielded improvement over the Majority baseline.

Pre-trained model. The Pre-trained TE Model slightly outperformed the LM on the word-order metric, suggesting its ability in determining semantic equivalence of the nominalization with the input sentence, a factor that is not considered by the LM. However, it is less sensitive to the choice of POS, prepositions and determiners to optimize fluency, as reflected by its lower POS and nominalization accuracy.

Fine-tuned model. Despite the limited size of our dataset, fine-tuning resulted in the strongest performance. In both settings and in terms of all three metrics, it produced statistically significant improvement over all other models.⁸ The improvement over the pre-trained version suggests that the semantic nuances that distinguish between the nominalization candidates are more subtle than the premises and hypotheses in general domains.

The performance was slightly lower if the TE model was used in the reverse direction, i.e., with the input sentence as hypothesis and its nominalized form as premise. This might be attributable to

⁸ $p < 0.05$ according to McNemar’s Test in both settings and all three metrics

the fact that the nominalized form is usually less specific in terms of tense and aspect.

7 Conclusion

This paper has reported the first quantitative evaluation on automatic clause nominalization and contributed the first dataset for this task. We have shown that a fine-tuned textual entailment model, followed with reranking with a language model, outperforms a number of competitive unsupervised approaches. In future work, we plan to extend our algorithm to determine whether a target clause can or should be nominalized, and to explore a richer variety of nominalization types.

Acknowledgements

We gratefully acknowledge support from a CityU Strategic Research Grant (project #7005709).

References

- Stephen Bailey. 2011. *Academic Writing: A Handbook for International Students*. Routledge, New York, NY.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding. In *Proc. North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL-HLT)*.
- Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243, Springer, Dordrecht.
- Atsushi Fujita and Satoshi Sato. 2008. A Probabilistic Model for Measuring Grammaticality and Similarity of Automatically Generated Paraphrases of Predicate Phrases. In *Proc. 22nd International Conference on Computational Linguistics (COLING)*.
- Nizar Habash and Bonnie Dorr. 2003. A Categorical Variation Database for English. In *Proc. North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL)*.
- Barbara Kamler and Pat Thomson. 2006. *Helping doctoral students write: Pedagogies for supervision*. Routledge, London.
- Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the Factual Consistency of Abstractive Text Summarization. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, page 9332–9346.
- Maria Lapata. 2002. The Disambiguation of Nominalizations. *Computational Linguistics*, 28(3):357–388.
- John S. Y. Lee, Ho Hung Lim, and Carol Webster. 2021. Paraphrasing Compound Nominalizations. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- John S. Y. 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.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- 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*.
- Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A New Benchmark for Natural Language Understanding. In *Proc. 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Sebastian Padó, Marco Pennacchiotti, and Caroline Sporleder. 2008. Semantic role assignment for event nominalisations by leveraging verbal data. In *Proc. 22nd International Conference on Computational Linguistics (COLING)*.
- Dariush Saberi and John Lee. 2019. Noun Generation for Nominalization in Academic Writing. In *Proc. Workshop on Computational Creativity in Natural Language Generation (CCNLG)*.
- Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena, and Eneko Agirre. 2021. Label Verbalization and Entailment for Effective Zero- and Few-Shot Relation Extraction. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, page 1199–1212.
- Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked Language Model Scoring. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Yusuke Shinyama, Satoshi Sekine, and Kiyoshi Sudo. 2002. Automatic Paraphrase Acquisition from News Articles. In *Proc. Second international conference on Human Language Technology Research (HLT)*.
- Harsh Trivedi, Heeyoung Kwon, Tushar Khot, Ashish Sabharwal, and Niranjan Balasubramanian. 2019. Repurposing Entailment for Multi-Hop Question Answering Tasks. In *Proc. NAACL-HLT*.