

2021

Supersense and Sensibility: Proxy Tasks for Semantic Annotation of Prepositions

Luke Gessler

Georgetown University, lukegessler@gmail.com

Shira Wein

Georgetown University, sw1158@georgetown.edu

Nathan Schneider

Georgetown University, nathan.schneider@georgetown.edu

Follow this and additional works at: <https://scholarworks.umass.edu/scil>



Part of the [Computational Linguistics Commons](#)

Recommended Citation

Gessler, Luke; Wein, Shira; and Schneider, Nathan (2021) "Supersense and Sensibility: Proxy Tasks for Semantic Annotation of Prepositions," *Proceedings of the Society for Computation in Linguistics: Vol. 4*, Article 59.

DOI: <https://doi.org/10.7275/xygw-fr77>

Available at: <https://scholarworks.umass.edu/scil/vol4/iss1/59>

This Paper is brought to you for free and open access by ScholarWorks@UMass Amherst. It has been accepted for inclusion in Proceedings of the Society for Computation in Linguistics by an authorized editor of ScholarWorks@UMass Amherst. For more information, please contact scholarworks@library.umass.edu.

Supersense and Sensibility: Proxy Tasks for Semantic Annotation of Prepositions

Luke Gessler Shira Wein Nathan Schneider

Georgetown University

{lg876, sw1158, nathan.schneider}@georgetown.edu

Introduction¹ Prepositions are highly ambiguous function words which can express a wide variety of relationships (Litkowski and Hargraves, 2006; Tratz, 2011). Supersenses have been proposed as an analytic framework for studying their lexical semantics, but extant gold-annotated corpora (e.g. Schneider et al., 2018) are small because preposition supersense annotation is a complex annotation task that requires much training and time.

Here, we present two **proxy** task designs for crowdsourcing from which supersense labels can be sensed – that is, recovered indirectly. These designs involve in-context **substitution** and **similarity** judgments. Based on four in-house pilot experiments, we conclude that both designs are promising methods for building a large preposition supersense-annotated corpus, and that they differ in difficulty for the annotators and for the researchers.

Prepositional Supersenses Prepositions can express many different kinds of semantic relations. Schneider et al. (2018) present SNACS, an annotation framework for prepositions encoding these relations. The meanings of prepositions are expressed in terms of supersenses, of which there are 50 in SNACS v2.5. For instance, the preposition *in* can be used to express time, place, and other relations: “I rented an apartment in_{LOCUS} Boston”, “I hope to see you in_{TIME} the future”.

Task 1: Preposition Substitution This design consists of two crowdsourced tasks and requires an unlabeled corpus \mathcal{U} . First, in the **substitute generation task**, we identify an unlabeled instance $\langle s, t \rangle \in \mathcal{U}$, where s is a sentence and $t \in s$ is the **target preposition** to be disambiguated. A crowdworker provides a substitute t' for t which approximately

¹This work is an abstract of work presented at LAW XIV. Please refer to the full paper for more detail, and prefer to cite the full paper over this abstract: <https://www.aclweb.org/anthology/2020.law-1.11/>

Figure 1: An example instance for the substitute selection task.

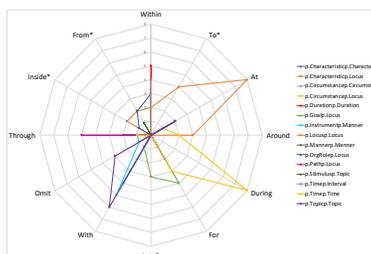


Figure 2: Substitutes for “in” selected in the substitute selection task. Each “spoke” is a substitute, and every point on each colored line represents the frequency of a substitute used for an instance that was gold-labeled with a particular supersense tag.

preserves the meaning of s when substituted with t and does not contain t . E.g., for the sentence “The book is **by** the lamp”, “close to” and “near” would both be good substitutes because “The book is **close to** the lamp” and “The book is **near** the lamp” both have similar meanings. By the end of this task, each several potential substitutes t'_1, \dots, t'_n will have been proposed by workers, but this data alone is not enough to infer a supersense for t in s .

More information is collected in the second task, the **substitute selection task**. The substitutes from the generation task t'_1, \dots, t'_n populate a multiple-choice list, and crowdworkers choose all items on the list which are acceptable substitutes for t in s . Once enough crowdworkers have completed the

Figure 3: An example instance for the neighbor selection task.

Case	Tagger	Crowd	“None”
1 (Tagger correct, gold present)	17/17	17/17	0/12
2 (Tagger incorrect, gold present)	0/12	6/12	5/12
3 (Tagger correct, gold absent)	3/3	0/3	2/3
4 (Tagger incorrect, gold absent)	0/8	0/8	5/8

Table 1: Supersense tagger accuracy and crowd accuracy for a pilot study with 40 instances, which were deliberately selected such that the tagger got 20 correct and 20 incorrect predictions on them. (The tagger’s real performance is closer to 85%.) Each row in this table represents instances grouped by whether the tagger correctly predicted the target’s gold tag and whether the target’s gold tag was present among the 5 neighbors, yielding four possibilities. (For example, the second row covers the instances where the tagger incorrectly predicted the supersense tag, but the target’s gold tag was present among the neighbors.) The “None” column indicates how many times “None” was chosen by the crowd.

selection task, we have a frequency distribution over the substitutes. These distributions can then be used to train classifiers to predict supersenses.

In pilot studies, we carry out both subtasks for five common prepositions: *for*, *with*, *to*, *from*, and *in*. Annotators were shown the instances and asked to write a single substitution per instance. Although our dataset was too small, we qualitatively evaluate our pilot data and find that we could expect a classifier to achieve good results with a larger quantity of similar data – see Figure 2.

Task 2: Neighbor Selection This design consists of a single task and requires a labeled corpus \mathcal{L} , an unlabeled input corpus \mathcal{U} , and some similarity function $sim(x, y)$ that can compare two unlabeled instances $\langle s_1, t_1 \rangle, \langle s_2, t_2 \rangle$ represent how similar the usages of t_1 and t_2 are as a real number.

An unlabeled instance $\langle s, t \rangle \in \mathcal{U}$ is selected, the *target* instance. sim is used to compare it to every instance in \mathcal{L} , and the top k most similar instances in \mathcal{L} are retrieved with their labels, $\langle s_1, t_1, \ell_1 \rangle, \dots, \langle s_k, t_k, \ell_k \rangle$. We call these retrieved instances the target’s *neighbors*. Neighbors may optionally be filtered, e.g. to ensure that no label ℓ is represented more than once among ℓ_1, \dots, ℓ_k .

The target sentence s is presented to crowdworkers along with s_1, \dots, s_k from the neighbors, and crowdworkers are asked to select any neighbors for which the usage of the preposition t_i in s_i most resembles the usage of t in s . For example, if the target sentence were “I was booked **at** the hotel”, “There was no cabbage **at** the store” would be a good neighbor to choose, while “My technician arrived **at** 11 pm” would not be good to choose. The predicted supersense tag is taken from the neighbor sentence selected most often by crowdworkers.

In our pilot studies, we use the same five prepositions from before and implement our *sim* function as cosine similarity between supersense tag class probability vectors from the system of Liu et al. (2020). We find that using this methodology, human workers are able to outperform the supersense tagger: in easy cases, the humans agree with the tagger; when the tagger makes a mistake but still succeeds in retrieving a neighbor with the correct supersense, humans usually select the correct neighbor; when no good option is available, humans tend to select “None”. See Table 1.

Conclusion We have presented two designs for deriving prepositional supersense tags from crowdsourced tasks, and we have investigated their efficacy through pilot studies, finding both promising for producing high-quality annotations. We have made idealizations throughout this work: all data was homogeneous with respect to genre, and crowdworkers had some knowledge of the prepositional supersense annotation guidelines which likely made them better than real-world crowdworkers. Moreover, we studied only 5 common prepositions covering 20 or so supersenses out of SNACS’s 50. In future work, we will implement these designs on crowdsourcing platforms to further investigate these designs’ efficacy and the extent to which these idealizations affect our results.

References

- Kenneth C. Litkowski and Orin Hargraves. 2006. [Coverage and inheritance in the preposition project](#). In *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions*.
- Nelson F. Liu, Daniel Hershcovich, Michael Kranzlein, and Nathan Schneider. 2020. [Lexical semantic recognition](#). *arXiv:2004.15008 [cs]*. ArXiv: 2004.15008.
- Nathan Schneider, Jena D. Hwang, Vivek Srikumar, Jakob Prange, Austin Blodgett, Sarah R. Moeller,

Aviram Stern, Adi Bitan, and Omri Abend. 2018. *Comprehensive Supersense Disambiguation of English Prepositions and Possessives*. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 185–196, Melbourne, Australia. Association for Computational Linguistics.

Stephen Tratz. 2011. *Semantically-enriched parsing for natural language understanding*. Ph.D., University of Southern California, United States – California. ISBN: 9781267076816.