University *of Ljubljana*
Faculty *of Computer and Information Science*

Extending SloWNet using Machine Translation and Digital Dictionary Database

Miha Bombek, Luka Dragar, David Ocepek

Abstract

For our project, we extended the current SloWNet 3.1 (43,460 synsets) by translating missing synsets from the Princeton Wordnet (117,000 synsets) using the pre-trained Nvidia NeMo sequence-to-sequence transformer. We then linked the extended SloWNet synsets to the Slovenian Digital Dictionary Database (DDD) thereby adding additional information such as use cases as well as additional words. Our current main problem is low synset translation precision which we intend to improve either by using DDD to detect incorrect translations or by improving our translation by adding additional context to our NeMo transformer.

Keywords
WordNet, machine translation, entity linking

Advisors: Assist. Prof. Dr. Slavko Žitnik, Assist. Aleš Žagar

Introduction

Wordnets are lexical knowledge graphs (KG) with many practical use cases, such as information retrieval, automatic text classification, machine translation (MT), etc. Despite their wide range of use cases, Wordnet construction is a difficult task often requiring manual annotation of tens of thousands of words by experts. Because of the aforementioned difficulties Wordnets are frequently built using MT models [2, 5]. For our project, we focused on extending the Slovene Wordnet (SloWNet) by translating and adding the synsets from the English Princeton Wordnet (PWN). We then added additional information, obtained from the DDD [1], such as use cases to each synset. Under our proposed approach one can potentially almost triple the number of synsets in SloWNet while simultaneously using DDD for anomaly detection and quality control therefore utilizing already existing manually annotated Slovene resources to achieve a construction accuracy that is likely to be above that of heuristic methods that are used to construct Wordnets for low resource languages.

Methodology

SloWNet representation

We used the newest implementation of SlowNet (version 3.1) [2] as the basis for our project, keeping already present translations and only adding new translations generated by the NeMo model. SlowNet is primarily stored as an XML document which is hard to use and unintuitive, therefore we converted it to graph form. Each node represented a synset and contained the following information: PWN synset id, list of English literals that are part of the synset, and list of Slovene literals (if any) that are part of the synset. Edges represented the relationships between synsets such as whether two synsets were similar in meaning, had opposite meanings or whether one synset was a hyponym to another synset.

Machine translation

For translating synsets from the English WordNet to Slovene, we utilized the NeMo toolkit [3], developed by NVIDIA. For our purposes, we used a pre-trained model for Slovene-English translation [4], which was trained on approximately 33 million translation pairs. We implemented both single and list-based synset translation, translating each synset word individually or all of them together as a comma-separated list of literals. The latter proved to be more accurate than the prior in instances where it was applicable (where a synset included multiple English literals) likely because it provided context to the NeMo model. Additionally, we put each individual literal into single quotation marks as otherwise, the NeMo model exhibited weird and undesirable behavior such as adding arbitrary verbs to adjectives and translating literals in different declensions. Both problems are still present but their frequency was significantly diminished by the above approach.

DDD Linking

To extend SloWNet with data from DDD, we first linked synsets from SloWNet to corresponding senses from DDD.

Firstly, we exported relevant data from DDD into a separate knowledge base. For each sense, we exported the category and the lemmatized form of its lexical unit. We also included the forms of lexical units of related senses. These related senses include synonyms, antonyms, hypernyms and hyponyms. The new knowledge base contains data for 199079 senses from DDD.

We then iterated over all synsets from SloWNet and for each synset, we iterate over all of its literals. For every literal, we choose senses whose lexical units have the same form as the literal, as possible candidates. For each candidate, we then check its synonyms. If at least one synonym matches at least one literal from the synset, we link the candidate sense with the synset.

Evaluation

We evaluated our extended SloWNet manually, as follows: firstly we randomly sampled 30 synsets, then for each Slovene literal we decided whether the literal matched the concept of the synset it was assigned too. We then calculated the precision as the ratio of literals that we consider to accurately represent that concept (synset) relative to the number of all Slovene literals assigned to that concept. Using the calculated precisions for the set of sampled concepts we then calculated the bootstrapped mean and standard deviation of our expected precision (Eq. 1).

$$\forall s \in S : \text{precision}(s) = \frac{\textit{words belonging to synset } s}{\textit{words assigned to synset } s},$$
$$\forall \hat{S} \in \text{bootstrap(S)} : \hat{\mu}_{\text{precision}} = \frac{1}{|\hat{S}|} \sum_{s \in \hat{S}} \text{precision}(s),$$
$$\mu_{\text{precision}} = E[\hat{\mu}_{\text{precision}}], \quad \sigma_{\text{precision}} = D[\hat{\mu}_{\text{precision}}]. \qquad (1)$$

Results

Comparison to SlowNet 3.1

The overall mean precision of our extended SloWNet was $\mu = 0.56$ with a standard deviation of $\sigma = 0.46$ (Fig. 1), however, it is important to note the precision of our machine-translated synsets was only $\mu = 0.37$, $\sigma = 0.42$. This is significantly lower than the manual translation precision of $\mu = 0.81$, $\sigma = 0.36$, therefore it is most likely necessary to further improve translation accuracy in order for our extended SlowNet to be usable in practice.

DDD linking analysis

With our linking method, we obtained 29045 links between DDD senses and SloWnet synsets. These contain 17135 distinct senses, linked to 8675 distinct synsets. It was expected, that on average, several senses would be linked to the same synset, because of the data structure in DDD. Each literal from the synsets should have its own sense in DDD, so synsets can contain several senses.

However in theory, the number of links should be the same as the number of distinct DDD senses linked, since each sense should be linked to at most one synset. This is not the case because of several factors. The most prevalent one is, that not all senses of lexical units in DDD are manually checked. There exist so called *dummy* senses, that are a result of automatic data imports. Data, that should be distributed among several different senses is packed into one *dummy* sense, before it is manually checked and distributed. This means that a *dummy* sense can contain synonyms from different senses, thus it can match with different synsets.

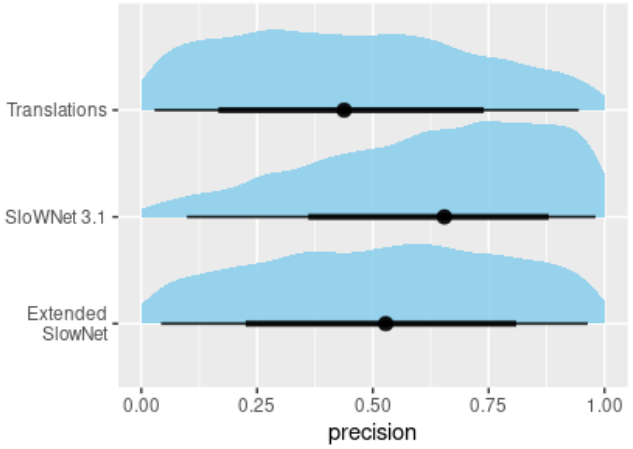

Figure 1. Precision of the SlowNet 3.1, our translations, and the extended SlowNet.

Future work & ideas

Improving synset translations

We have analyzed two additional approaches for improving the synset translation. The first would be to translate the Slovene literals back into English and remove or flag literals that did not translate into the synset. The second approach would be to try to inject further context into our English literals by adding sentences and definitions to them from wiktionary. During implementation, we found that the first approach removed too many accurately translated literals presumably due to missing synonyms in PWN and we were unable to consistently extract translated literals in the second approach because our NeMo model frequently reorganizes more complex text.

Lematization of Slovene literals

English literals are occasionally translated into Slovene in non-standard declensions. While this does not seriously impact our precision it does lead to an increased multiplicity of certain literals and increases the difficulty of linking the DDD database with our SlowNet, therefore we intend to use a language processing package such as CLASSLA [7] to standardize (lematize) our translations.

Anomaly detection using DDD

By linking entities from DDD and SloWNet, we could better evaluate the quality of translations and maybe even improve translation accuracy. For example, if we get a link between

a DDD sense and a SloWnet synset by matching synonyms, there is a high probability, that the matching literals were translated correctly so that they still remained synonyms in Slovene in order to be matched with synonyms from DDD. With this approach, we could introduce some sort of probability, that a translation is correct for every literal.

We could also remove literal translations in linked synsets, that didn't match any of the synonyms from DDD. This would improve the overall accuracy of the translations, but we could lose some correct translations, due to them not being listed as synonyms in DDD.

Improving DDD linking

With our current linking method, we only link entities by matching them by their form and checking shared synonyms. We only link entities, that share at least one synonym. This means, that we disregard synsets with only a single literal since they have no synonyms to match.

For this purpose, we will also include matching by other sense relations, such as antonyms and hypernyms. There are much less of these relations in DDD, but they might help by linking to some single literal synsets. We will also link by part-of-speech tags, which should cover many more of the single literal synsets. However, we feel that this method will not be as accurate as matching by sense relations. Therefore we will also include some empirically determined probabilities for every link. A link, acquired by matching synonyms, antonyms, and hypernyms should have a higher probability of being correct, than a link, acquired only by matching part-of-speech tags.

Automatic evaluation

Word embeddings are a popular tool for comparing the similarity of words and can be adapted to validate the words within synsets. The method for converting a lexical semantic graph to lexical semantic space [6] was used in conjunction with a standard similarity prediction task to obtain the evaluation of newly constructed synsets. There is an analogy dataset available for Slovene [8] that could be used to evaluate our constructed embeddings. We note however that this method is only suitable for evaluating the synset validity but cannot evaluate the relations between them.

References

[1] CJVT. *Digital Dictionary Database*. Accessed on 16. 3. 2023. 2023. URL: https://wiki.cjvt.si/books/digital-dictionary-database.

[2] Darja Fišer and Tomaž Erjavec. *sloWNet: construction and corpus annotation*.

[3] Oleksii Kuchaiev et al. "NeMo: a toolkit for building AI applications using Neural Modules". In: *CoRR* abs/1909.09577 (2019). arXiv: 1909.09577. URL: http://arxiv.org/abs/1909.09577.

[4] Iztok Lebar Bajec et al. *Neural Machine Translation model for Slovene-English language pair RSDO-DS4-NMT 1.2.6*. Slovenian language resource repository CLARIN.SI. 2022. URL: http://hdl.handle.net/11356/1736.

[5] Roberto Navigli and Simone Paolo Ponzetto. "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network". In: *Artificial intelligence* 193 (2012), pp. 217–250.

[6] Chakaveh Saedi et al. "Wordnet embeddings". In: *Proceedings of the Third Workshop on Representation Learning for NLP* (2018). DOI: 10.18653/v1/W18-3016.

[7] Luka Terčon, Jaka Čibej, and Nikola Ljubešić. *The CLASSLA-Stanza model for lemmatisation of standard Slovenian 2.0*. Slovenian language resource repository CLARIN.SI. 2023. URL: http://hdl.handle.net/11356/1768.

[8] Matej Ulčar et al. "Multilingual Culture-Independent Word Analogy Datasets". In: *Proceedings of the Twelfth Language Resources and Evaluation Conference* (2020).