Practical Approach on Implementation of WordNets for South African Languages

Tshephisho Joseph Sefara
Council for Scientific and Industrial Research
Pretoria, South Africa
tsefara@csir.co.za

Tumisho Billson Mokgonyane
Department of Computer Science
University of Limpopo, South Africa
tumisho.mokgonyane@ul.ac.za

Vukosi Marivate
Department of Computer Science
University of Pretoria, South Africa
vukosi.marivate@cs.up.ac.za

Abstract

This paper proposes the implementation of WordNets for five South African languages, namely, Sepedi, Setswana, Tshivenda, isiZulu and isiXhosa to be added to open multilingual WordNets (OMW) on natural language toolkit (NLTK). The African WordNets are converted from Princeton WordNet (PWN) 2.0 to 3.0 to match the synsets in PWN 3.0. After conversion, there were 7157, 11972, 1288, 6380, and 9460 lemmas for Sepedi, Setswana, Tshivenda, isiZulu and isiXhosa respectively. Setswana, isiXhosa, Sepedi contains more lemmas compared to 8 languages in OMW and isiZulu contains more lemmas compared to 7 languages in OMW. A library has been published for continuous development of African WordNets in OMW using NLTK.

1 Introduction

WordNet consists of information about adverbs, adjectives, verbs and nouns in English and it organizes the words according to the notion of a synset. A synset can be defined as a set of words that are interchangeable in certain context. For example, the set {house, home, building} form a synset since the words can be used interchangeably referring to the same concept. Synsets can be linked to each other by means of semantic relations such as meronymy (leaf-tree), hypernymy versus hyponymy relation (flower-rose). The interlinked synsets create a strong semantic network that allows researchers (a) to automatically expand their search queries in information retrieval tasks (Azad and Deepak, 2019; Abbache et al., 2016), (b) to artificially expand their dataset by making use of data augmentation in natural language processing (NLP) tasks (Marivate and Sefara, 2020), (c) to improve cybercrime investigation in social network mining tasks (Iqbal et al., 2019).

WordNets have been applied in many domains such as machine learning classification to improve the performance of classification algorithms. An example is the role of WordNets is to increase the amount of NLP task data via data augmentation (Marivate and Sefara, 2020). This has been done for many well-resourced (European) languages. For low-resource languages such as South African languages, few studies have been done to build WordNets under low resources. African WordNets (Bosch and Griesel, 2017; Griesel et al., 2019) is a project that develop aligned WordNets for African languages spoken in South Africa. Initially, the project included five languages Sepedi, Setswana, Tshivenda, isiXhosa and isiZulu. During the development, DEBVisDic (WordNet editor) was used to build semantic networks. Due to limited resources, the expand model was followed during the development of the African WordNets. The expand model in WordNets creation is when the structure of Princeton WordNet is used to create other WordNets in other languages.

The goal of this paper is to build a multilingual lexical database with WordNets for South African languages based on the Princeton WordNet 3 to be utilized using the Python NLTK[1] library via open multilingual WordNets (OMW)[2]. We utilize the WordNets previously built by Bosch and Griesel (2017) for Sepedi, Setswana, Tshivenda, isiXhosa and isiZulu, to be compatible with OMW standard.

The main contributions of this paper are as fol-

[1] http://www.nltk.org/

[2] http://compling.hss.ntu.edu.sg/omw/

lows:

- A Python library has been released to allow inspection and improvement of the resource. The library can be found on Github[3] and Python repository[4].
- We released and published the data set (Sefara et al., 2020).

The outline of this paper is as follows: In Section 2 we discuss literature review of WordNets and their applications. Section 3 describes the methodology taken to create the resource. Section 4 concludes the paper with future work.

2 Literature Review

This section discusses the current WordNets and their applications in various domains.

2.1 WordNets

Bond and Foster (2013) created OMW that support more than 150 languages. OMW is made by combining WordNets published with open source licenses, Wiktionary data, and Unicode Common Locale Data Repository. The aim of OMW is to provide access to WordNets in multiple languages. All the WordNets in OMW are linked to PWN (Miller, 1995). The OMW and PWN can be accessed through NLTK.

EuroWordNet is a project created by Vossen (1998) to build multilingual WordNets for European languages based on PWN. The goal of EuroWordNet is to create multilingual database, build WordNets independently, obtain compatibility across languages, and to maintain language-specific relations.

Postma et al. (2016) created an open WordNet for Dutch that contains a total of 117,914 synsets using data from Cornetto database, open source resources, and the PWN. Authors also created a Python module[5] that can be applied to NLP applications.

ElKateb et al. (2006) proposed the development of WordNet for Arabic language using PWN for English as basis. Authors constructed the Arabic WordNet by using methods used to develop the EuroWordNet (Vossen, 1998). Regragui et al. (2016) added new content to Arabic WordNet that improved the performance of NLP applications such as question answering.

Bosch and Griesel (2017) discussed methods to build WordNets for low-resourced languages when the development of WordNets for South African languages was initiated using expand model based on PWN version 2. Authors created a total of 53982 synsets, 9279 definitions and 28853 usage examples. Due to low-resource environment, identification and translation of appropriate synsets was done by a human expert. One of the method is that authors used bilingual dictionaries to transfer information from dictionary to WordNet then a linguists make final approval for inclusion in the WordNets.

The Finnish WordNet is a lexical database for Finnish based on PWN structure (Lindén and Carlson, 2010). All word senses in PWN were translated into Finnish to make FinnWordNet. The PWN word senses were translated by a human translator to validate the quality of the content. The translation process is explained by (Lindén and Carlson, 2010). FinnWordNet has 117659 synsets and freely available under Creative Commons 3.0 license.

2.2 Applications of WordNets

Baccianella et al. (2010) annotated all the synsets of WordNet (Miller, 1995) with respect to the notions of positivity, negativity, and neutrality to create new dataset called SentiWordNet, an improved lexical resource that is designed to support opinion mining applications and sentiment classification. Authors published the dataset on Github[6].

Siddharthan et al. (2018) uses WordNet to create WordNet-feelings which is a new dataset that categorises word senses as feelings. Authors created ten categories and manually annotated the dataset by adding new categories and definitions.

WordNets are used as data sources in search and information retrieval tasks when building a query (Azad and Deepak, 2019). Abbache et al. (2016) improved the performance of information retrieval system for Arabic language by using WordNet and association rules to expand the search query. In their methodology, authors removed stop words (functional words) from the query before extracting and selecting synonyms using Arabic WordNet as the main source for word selection.

Marivate and Sefara (2020) used WordNet to

[3] https://github.com/JosephSefara/AfricanWordNet

[4] https://pypi.org/project/africanwordnet

[5] https://github.com/cltl/OpenDutchWordnet

[6] https://github.com/aesuli/SentiWordNet

create data augmentation technique for NLP classification applications. Authors compared the technique with semantic similarity augmentation and round-trip augmentation. The WordNet-based augmentation improved the performance of the classification models when using Wikipedia dataset. The same WordNet-based augmentation was used by Zhang et al. (2015) to train a temporal convolutional network that learns text understanding from character level input up to an abstract text concepts. Hasan et al. (2020) applied semantic similarity of WordNet to manage the ambiguity in social media text by selecting informative features to enhance semantic representation.

3 Methodology

This section discusses the design and implementation of the WordNets for South African languages. It first discusses sense map preparation, then WordNets conversion, and finally implementation.

3.1 Sense map preparation

In this section, we explain the sense map preparation process.

We used the sensemap(5WN) published on PWN website [7] for versions 2.0, 2.1, and 3.0. Sense map simply list each 2.0 noun sense (encoded as a sense key) paired with its mapping to one or more 2.1 noun senses. We converted all the polysemous (nouns and verbs) and monosemous (nouns and verbs) to 2.1. Then lastly, we converted 2.1 synsets to 3.0. We used the 3.0 sense maps to convert all the synsets from 2.1 to 3.0. The synsets that are not in all the sense maps are used as is.

Algorithm 1 illustrates the steps taken during conversion of the sense maps. The algorithm was run twice, for first time to convert 2.0→2.1 then duplicate offset targets in 2.1 were removed. The second time to convert 2.1→3.0 then duplicate offset target in 3.0 were also removed. Table 1 shows a sample of the converted offsets that will later be used to match every synset in African WordNets from 2.0 to 3.0.

Table 1 shows a sample format of sense mapping that are later used to convert the WordNets from 2.0 to 3.0.

Algorithm 1: Sense map conversion

```
Input: $s$: sense map file
Output: $\hat{s}$ list containing pair of source
        and target offset ID
def mono ($s$):
    Let $F \leftarrow Open(s)$ be a to file reader;
    for $line$ in $F$:
        $SourceOffset \leftarrow$ use regular
          expression to match source offset
          ID from $line$;
        $TargetOffset \leftarrow$ use regular
          expression to match target offset
          ID from $line$;
        $\hat{s} \leftarrow [Sourceoffset,$
          $Targetoffset]$;
    return($\hat{s}$);
```

Table 1: Sample of the sense mapping

2.0	2.1	3.0
12976279-n	13571065-n	13752172-n
12976532-n	13571318-n	13752443-n

3.2 WordNets conversion

This section discusses conversion of African WordNets to PWN 3.0 and explain OMW format.

We collected the WordNets created by Bosch and Griesel (2017) from South African Centre for Digital Language Resources (SADiLaR)[8]. SADiLaR is a national center supported and funded by the South African Department of Science and Innovation. The WordNets are in the form of XML format based on PWN version 2.

We used a library called BeautifulSoup[9] to extract all the synset offset ID, part-of-speech tag, lemma, and word form since the WordNets are in XML format. Table 2 shows the number of synsets before and after conversion to 3.0 excluding the synsets that do not exists in PWN. There is an increase in number of synsets, isiZulu increased by 90, isiXhosa by 150, Sepedi by 101, Setswana by 240, and Tshivenda by 16. The increase is caused by synsets that have multiple mappings in PWN 3.0. We saved the new synsets in a format that is supported by OMW. The OMW format is as follows:

offset-pos langcode:lemma wordform

[7] https://wordnet.princeton.edu/documentation/senseidx5wn

[8] https://www.sadilar.org/

[9] https://pypi.org/project/beautifulsoup4/

where *offset* is the unique ID (linking to the PWN), *langcode* is the universal language code[10], *wordform* is the written word, and *pos* is the part-of-speech.

Table 2: Conversion of synsets

Language	Original Synsets	New Synsets
isiZulu	9026	9116
isiXhosa	13731	13881
Sepedi	10647	10748
Setswana	22234	22474
Tshivenda	1581	1597

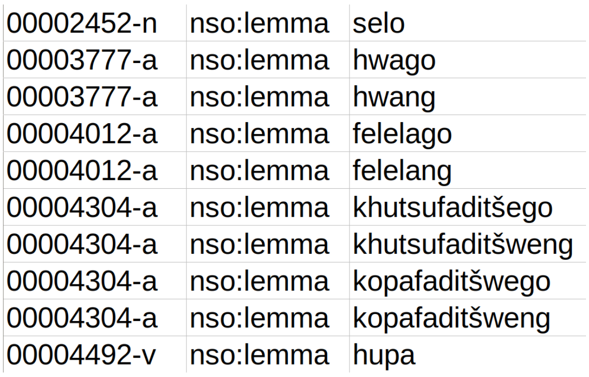
An example of the formatted synsets is depicted in Figure 1 that is compatible with OMW in NLTK. OMW consists of 29 languages in NLTK as shown in Table 3. There are 8 languages in OMW that contains lemmas less than that of Setswana, isiXhosa, and Sepedi. IsiZulu contains more lemmas than 7 languages in OMW while Tshivenda contains the smallest lemmas than all other languages.

Table 3: OMW in NLTK

Language	Lemma	Language	Lemma
eng	147306	glg	23124
fin	129839	ell	18225
jpn	89637	arb	17785
tha	80508	fas	17560
cmn	61532	**tsn**	**11972**
fra	55350	**xho**	**9460**
por	54069	**nso**	**7157**
cat	46531	bul	6720
pol	45387	**zul**	**6380**
nld	43077	als	5988
ita	41855	swe	5824
slv	41032	heb	5325
ind	36954	dan	4468
spa	36681	nob	4186
zsm	33932	nno	3387
hrv	29010	qcn	3206
eus	26240	**ven**	**1288**

3.3 Implementation

This section discuss implementation of African WordNets in NLTK.



00002452-n	nso:lemma	selo
00003777-a	nso:lemma	hwago
00003777-a	nso:lemma	hwang
00004012-a	nso:lemma	felelago
00004012-a	nso:lemma	felelang
00004304-a	nso:lemma	khutsufaditšego
00004304-a	nso:lemma	khutsufaditšweng
00004304-a	nso:lemma	kopafaditšwego
00004304-a	nso:lemma	kopafaditšweng
00004492-v	nso:lemma	hupa

Figure 1: An extract of the converted WordNet for Sepedi using OMW format

Total of 5 files (sample shown in Figure 1) have been created that consists of the 5 languages to be added to NLTK. The files have been named according to the following format:

- Sepedi: wn-data-nso.tab
- Setswana: wn-data-tsn.tab
- isiXhosa: wn-data-xho.tab
- isiZulu: wn-data-zul.tab
- Tshivenda: wn-data-ven.tab

where each file resides in a directory inside OMW corpus in NLTK and the directory name is named according to the ISO language code. The ISO language code for Sepedi is **nso**, Setswana is **tsn**, isiXhosa is **xho**, isiZulu is **zul**, and Tshivenda is **ven**.

A Python helper library[11] has been created to install these African WordNets to OMW in NLTK. The African WordNets can be used like other WordNets on OMW. For example, the library has to be imported to the environment then the following statements shows the lemma names of the word ’entity’ in Setswana:

```
>>> from nltk.corpus import wordnet
>>> import africanwordnet
>>> wordnet.synset(’entity.n.01’).lemmas(’tsn’)
[Lemma(’entity.n.01.selô’),
Lemma(’entity.n.01.sengwe’)]
```

Listing 1: Lemma example

The following statement is used to view the synsets of the isiZulu word ’iqoqo’ (means collection).

```
>>> from nltk.corpus import wordnet
>>> import africanwordnet
>>> wordnet.synsets(’iqoqo’,lang=(’zul’))
[Synset(’whole.n.02’),
 Synset(’conspectus.n.01’),
```

[10] https://www.loc.gov/standards/iso639-2/php/code_list.php

[11] https://pypi.org/project/africanwordnet

```
Synset(’overview.n.01’),
Synset(’sketch.n.03’),
Synset(’compilation.n.01’),
Synset(’collection.n.01’),
Synset(’team.n.02’),
Synset(’set.n.01’)]
```

Listing 2: Synonym example

The following statement is used to view the hyponyms of the Sepedi word ’taelo’ (means edict).

```
>>> from nltk.corpus import wordnet
>>> import africanwordnet
>>> synsets = wn.synsets(’taelo’,lang=(’nso’))
>>> for synset in synsets:
...     for hypo in synset.hyponyms():
...         for lemma in hypo.lemmas(”nso”):
...             print(lemma)
Lemma(’behest.n.01.tlhalošo’)
Lemma(’commandment.n.01.molao’)
Lemma(’commandment.n.01.taelo’)
Lemma(’commission.n.06.taelo’)
Lemma(’injunction.n.01.taelo’)
Lemma(’order.n.01.taelo’)
Lemma(’summons.n.02.tagafalo’)
```

Listing 3: Hyponym example

The following statement is used to view the hypernyms of the isiXhosa word ’omisa’ (means dry).

```
>>> from nltk.corpus import wordnet
>>> import africanwordnet
>>> synsets = wn.synsets(’omisa’,lang=(’xho’))
>>> for synset in synsets:
...     for hypo in synset.hypernyms():
...         for lemma in hypo.lemmas(”xho”):
...             print(lemma)
Lemma(’dry.v.01.omisa’)
Lemma(’change.v.01.guqula’)
Lemma(’change.v.01.tshintsha’)
Lemma(’change_integrity.v.01.guqula_imfezeko’)
```

Listing 4: Hypernyms example

4 Conclusion and Future Work

This paper presented the implementation of African WordNets to be used in NLTK via OMW. We discussed the conversion of PWN sense maps from 2.0 to 2.1 to 3.0. There was an increase of synsets during conversion. We proposed an algorithm that helps to convert synsets from PWN 2.0 to 3.0. A Python library has been made available[12] to utilize the WordNets.

The future work will focus on

- improving conversion of PWN sense maps from 2.0 to 3.0 so that all synsets are available in 3.0. Kim et al. (2018) proposed automatic mapping of synsets using bilingual dictionaries. Due to limited bilingual dictionaries this method could not be utilized.
- evaluation of the African WordNets using various evaluation methods. Ramanand and Bhattacharyya (2007) proposed a method to evaluate synsets using dictionary definitions since currently there are no enough dictionaries for these languages this method could not be utilized.

Acknowledgments

We would like to thank SADILAR for making the datasets available for research. And we would like to thank Github for hosting the developed library.

References

Ahmed Abbache, Farid Meziane, Ghalem Belalem, Fatma Zohra Belkredim, et al. 2016. Arabic query expansion using WordNet and association rules. *International Journal of Intelligent Information Technologies (IJIIT)*, 12(3):51–64.

Hiteshwar Kumar Azad and Akshay Deepak. 2019. A new approach for query expansion using Wikipedia and WordNet. *Information sciences*, 492:147–163.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In *Lrec*, volume 10, pages 2200–2204.

Francis Bond and Ryan Foster. 2013. Linking and extending an open multilingual wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1352–1362.

Sonja E Bosch and Marissa Griesel. 2017. Strategies for building wordnets for under-resourced languages: The case of african languages. *Literator (Potchefstroom. Online)*, 38(1):1–12.

Sabry ElKateb, William Black, Horacio Rodríguez, Musa Alkhalifa, Piek Vossen, Adam Pease, and Christiane Fellbaum. 2006. Building a wordnet for arabic. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC’06)*, pages 29–34.

Marissa Griesel, Sonja Bosch, and Mampaka L Mojapelo. 2019. Thinking globally, acting locally–progress in the African wordnet project. In *Proceedings of the Tenth Global Wordnet Conference*, pages 191–196.

Ali Muttaleb Hasan, Noorhuzaimi Mohd Noor, Taha Hussein Rassem, Shahrul Azman Mohd Noah, and Ahmed Muttaleb Hasan. 2020. A proposed method using the semantic similarity of wordnet 3.1 to handle the ambiguity to apply in social media text. In *Information Science and Applications*, pages 471–483. Springer.

[12] https://pypi.org/project/africanwordnet

Farkhund Iqbal, Benjamin CM Fung, Mourad Debbabi, Rabia Batool, and Andrew Marrington. 2019. Wordnet-based criminal networks mining for cybercrime investigation. *IEEE Access*, 7:22740–22755.

Jiseong Kim, Younggyun Hahm, Sunggoo Kwon, and Key-Sun Choi. 2018. Automatic wordnet mapping: from corenet to princeton wordnet. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Krister Lindén and Lauri Carlson. 2010. Finnwordnet-wordnet på finska via översättning. *LexicoNordica – Nordic Journal of Lexicography*, 17:119–140.

Vukosi Marivate and Tshephisho Sefara. 2020. Improving short text classification through global augmentation methods. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 385–399. Springer.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

MC Postma, E Miltenburg, R Segers, A Schoen, and PTJM Vossen. 2016. Open dutch wordnet. In *Proceedings of the Eigth Global Wordnet Conference*.

J Ramanand and Pushpak Bhattacharyya. 2007. Towards automatic evaluation of wordnet synsets. *GWC 2008*, page 360.

Yasser Regragui, Lahsen Abouenour, Fettoum Krieche, Karim Bouzoubaa, and Paolo Rosso. 2016. Arabic wordnet: New content and new applications. In *Proceedings of the Eighth Global WordNet Conference*, pages 330–338.

Tshephisho Sefara, Tumisho Mokgonyane, and Vukosi Marivate. 2020. Wordnets for South African languages. Zenodo, December.

Advaith Siddharthan, Nicolas Cherbuin, Paul J Eslinger, Kasia Kozlowska, Nora A Murphy, and Leroy Lowe. 2018. WordNet-feelings: a linguistic categorisation of human feelings. *arXiv preprint arXiv:1811.02435*.

Piek Vossen, 1998. *Introduction to EuroWordNet*, pages 1–17. Springer Netherlands, Dordrecht.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.