The risk of sub-optimal use of Open Source NLP Software: UKB is inadvertently state-of-the-art in knowledge-based WSD

Eneko Agirre
IXA NLP group
UPV/EHU

Oier López de Lacalle
IXA NLP group
UPV/EHU

Aitor Soroa
IXA NLP group
UPV/EHU

{e.agirre,oier.lopezdelacalle,**a.soroa**}@ehu.eus

Abstract

UKB is an open source collection of programs for performing, among other tasks, knowledge-based Word Sense Disambiguation (WSD). Since it was released in 2009 it has been often used out-of-the-box in sub-optimal settings. We show that nine years later it is the state-of-the-art on knowledge-based WSD. This case shows the pitfalls of releasing open source NLP software without optimal default settings and precise reproducibility instructions.

1 Introduction

The release of open-source Natural Language Processing (NLP) software has been key to make the field progress, as it facilitates other researchers to build upon previous results and software easily. It also allows easier reproducibility, allowing for sound scientific progress. Unfortunately, in some cases, it can also allow competing systems to run the open-source software out-of-the-box with sub-optimal parameters, specially in fields where there is no standard benchmark and new benchmarks (or new versions of older benchmarks) are created.

Once a paper reports sub-optimal results for a NLP software, newer papers can start to routinely quote the low results from the previous study. Finding a fix to this situation is not easy. The authors of the software can contact the authors of the more recent papers, but it is usually too late for updating the paper. Alternatively, the authors of the NLP software can try to publish a new paper with updated results, but there is usually no venue for such a paper, and, even if published, it might not be noticed in the field.

In this paper we want to report such a case in Word Sense Disambiguation (WSD), where the original software (UKB) was released with sub-optimal default parameters. Although the accompanying papers did contain the necessary information to obtain state-of-the-art results, the software did not contain step-by-step instructions, or end-to-end scripts for optimal performance. This case is special, in that we realized that the software is able to attain state-of-the-art results also in newer datasets, using the same settings as in the papers.

The take-away message for open-source NLP software authors is that they should not rely on other researchers reading the papers with care, and that it is extremely important to include, with the software release, precise instructions and optimal default parameters, or better still, end-to-end scripts that download all resources, perform any necessary pre-processing and reproduce the results.

The first section presents UKB and WSD, followed by the settings and parameters. Next we present the results and comparison to the state-of-the-art. Section 5 reports some additional results, and finally, we draw the conclusions.

2 WSD and UKB

Word Sense Disambiguation (WSD) is the problem of assigning the correct sense of a word in a context (Agirre and Edmonds, 2007). Traditionally, supervised approaches have attained the best results in the area, but they are expensive to build because of the need of large amounts of manually annotated examples. Alternatively, knowledge based approaches rely on lexical resources such as WordNet, which are nowadays widely available in many languages (Bond and Paik, 2012)[1]. In particular, graph-based approaches represent the knowledge base as a graph, and apply several well-known graph analysis algorithms to perform WSD.

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

Proceedings of Workshop for NLP Open Source Software, pages 29–33
Melbourne, Australia, July 20, 2018. ©2018 Association for Computational Linguistics

UKB is a collection of programs which was first released for performing graph-based Word Sense Disambiguation using a pre-existing knowledge base such as WordNet, and attained state-of-the-art results among knowledge-based systems when evaluated on standard benchmarks (Agirre and Soroa, 2009; Agirre et al., 2014). In addition, UKB has been extended to perform disambiguation of medical entities (Agirre et al., 2010), named-entities (Erbs et al., 2012; Agirre et al., 2015), word similarity (Agirre et al., 2009) and to create knowledge-based word embeddings (Goikoetxea et al., 2015). All programs are open source[2,3] and are accompanied by the resources and instructions necessary to reproduce the results. The software is quite popular, with 60 stars and 26 forks in github, as well as more than eight thousand direct downloads from the website since 2011. The software is coded in C++ and released under the GPL v3.0 license.

When UKB was released, the papers specified the optimal parameters for WSD (Agirre and Soroa, 2009; Agirre et al., 2014), as well as other key issues like the underlying knowledge-base version, specific set of relations to be used, and method to pre-process the input text. At the time, we assumed that future researchers would use the optimal parameters and settings specified in the papers, and that they would contact the authors if in doubt. The default parameters of the software were not optimal, and the other issues were left under the users responsibility.

The assumption failed, and several papers reported low results in some new datasets (including updated versions of older datasets), as we will see in the following sections.

3 UKB parameters and setting for WSD

When using UKB for WSD, the main parameters and settings can be classified in five main categories. For each of those we mention the best options and the associated UKB parameter when relevant (in italics), as taken from (Agirre and Soroa, 2009; Agirre et al., 2014):

- Pre-processing of input text. When running UKB for WSD, one needs to define which window of words is to be used as context to initialize the random walks. One option is to take just the sentence, but given that in some cases the sentences are very short, better results are obtained when considering previous and following sentences. The procedure in the original paper repeated the extension procedure until the total length of the context is at least 20 words[4].

- Knowledge base relations. When performing WSD for English, UKB uses WordNet (Fellbaum, 1998) as a knowledge base. WordNet comes in various versions, and usually UKB performs best when using the same version the dataset was annotated with. Besides regular WordNet relations, gloss relations (relations between synsets appearing in the glosses) have been shown to be always helpful.

- Graph algorithm. UKB implements different graph-based algorithms and variants to perform WSD. These are the main ones:
  ppr_w2w: apply personalized PageRank for each target word, that is, perform a random walk in the graph personalized on the word context. It yields the best results overall, at the cost of being more time consuming that the rest.
  ppr: same as above, but apply personalized PageRank to each sentence only once, disambiguating all content words in the sentence in one go. It is thus faster that the previous approach, but obtains worse results.
  dfs: unlike the two previous algorithms, which consider the WordNet graph as a whole, this algorithm first creates a subgraph for each context, following the method first presented in Navigli and Lapata (2010), and then runs the PageRank algorithm over the subgraph. This option represents a compromise between *ppr_w2w* and *ppr*, as it faster than than the former while better than the latter.

- The PageRank algorithm has two parameters which were set as follows: number of iterations of power method (*prank_iter*) 30, and damping factor (*prank_damping*) 0.85.

- Use of sense frequencies (*dict_weight*). Sense frequencies are a valuable piece of informa-

[2] http://ixa2.si.ehu.eus/ukb

[3] https://github.com/asoroa/ukb

[4] The number 20 was initial arbitrarily set in the experiments of (Agirre and Soroa, 2009) somewhat arbitrarily, and never changed afterwards.

	All	S2	S3	S07	S13	S15
UKB (this work)	**67.3**	68.8	66.1	53.0	**68.8**	**70.3**
UKB (elsewhere)†‡	57.5	60.6	54.1	42.0	59.0	61.2
Chaplot and Sakajhutdinov (2018) ‡	66.9	**69.0**	**66.9**	55.6	65.3	69.6
Babelfy (Moro et al., 2014)†	65.5	67.0	63.5	51.6	66.4	70.3
MFS	65.2	66.8	66.2	55.2	63.0	67.8
Basile et al. (2014)†	63.7	63.0	63.7	**56.7**	66.2	64.6
Banerjee and Pedersen (2003)†	48.7	50.6	44.5	32.0	53.6	51.0

Table 1: F1 results for knowledge-based systems on the (Raganato et al., 2017a) dataset. Top rows show conflicting results for UKB. † for results reported in (Raganato et al., 2017a), ‡ for results reported in (Chaplot and Sakajhutdinov, 2018). Best results in bold. S2 stands for Senseval-2, S3 for Senseval-3, S07 for Semeval-2007, S13 for Semeval-2013 and S15 for Semeval-2015.

	All	S2	S3	S07	S13	S15
Yuan et al. (2016)	**71.5**	**73.8**	**71.8**	63.5	**69.5**	**72.6**
Raganato et al. (2017b)	69.9	72.0	69.1	**64.8**	66.9	71.5
Iacobacci et al. (2016)†	69.7	73.3	69.6	61.1	66.7	70.4
Melamud et al. (2016)†	69.4	72.3	68.2	61.5	67.2	71.7
IMS (Zhong and Ng, 2010)†	68.8	72.8	69.2	60.0	65.0	69.3

Table 2: F1 results for supervised systems on the (Raganato et al., 2017a) dataset. † for results reported in (Raganato et al., 2017a). Best results in bold. Note that (Raganato et al., 2017b) used S07 for development.

tion that describe the frequencies of the associations between a word and its possible senses. The frequencies are often derived from manually sense annotated corpora, such as Semcor (Miller et al., 1993). We use the sense frequency accompanying Wordnet, which, according to the documentation, ”represents the decimal number of times the sense is tagged in various semantic concordance texts”. The frequencies are smoothed adding one to all counts (*dict_weight_smooth*). The sense frequency is used when initializing context words, and is also used to produce the final sense weights as a linear combination of sense frequencies and graph-based sense probabilities. The use of sense frequencies with UKB was introduced in (Agirre et al., 2014).

4 Comparison to the state-of-the-art

We evaluate UKB on the recent evaluation dataset described in (Raganato et al., 2017a). This dataset comprises five standard English all-words datasets, standardized into a unified format with gold keys in WordNet version 3.0 (some of the original datasets used older versions of WordNet). The dataset contains $7,253$ instances of $2,659$ different content words (nouns, verbs, adjectives and adverbs). The average ambiguity of the words in the dataset is of 5.9 senses per word. We report F1, the harmonic mean between precision and recall, as computed by the evaluation code accompanying the dataset.

The two top rows in Table 1 show conflicting results for UKB. The first row corresponds to UKB ran with the settings described above. The second row was first reported in (Raganato et al., 2017a). As the results show, that paper reports a suboptimal use of UKB. In more recent work, Chaplot and Sakajhutdinov (2018) take up that result and report it in their paper as well. The difference is of nearly 10 absolute F1 points overall.[5] This decrease could be caused by the fact that Raganato et al. (2017a) did not use sense frequencies.

In addition to UKB, the table also reports the best performing knowledge-based systems on this dataset. Raganato et al. (2017a) run several well-known algorithms when presenting their datasets. We also report (Chaplot and Sakajhutdinov, 2018),

[5] Note that the UKB results for S2, S3 and S07 (62.6, 63.0 and 48.6 respectively) are different from those in (Agirre et al., 2014), which is to be expected, as the new datasets have been converted to WordNet 3.0 (we confirmed experimentally that this is the sole difference between the two experiments).

	All	S2	S3	S07	S13	S15
Single context sentence						
ppr_w2w	66.9	**69.0**	65.7	53.9	67.1	69.9
dfs_ppr	65.2	67.5	65.6	53.6	62.7	68.2
ppr	65.5	67.5	**66.5**	**54.7**	63.3	67.4
ppr_w2w$_{\texttt{nf}}$	60.2	63.7	55.1	42.2	63.5	63.8
ppr$_{\texttt{nf}}$	57.1	60.5	53.8	41.3	58.0	61.4
dfs$_{\texttt{nf}}$	58.7	63.3	52.8	40.4	61.6	62.5
One or more context sentences ($\#words \geq 20$)						
ppr_w2w	**67.3**	68.8	66.1	53.0	**68.8**	**70.3**
ppr	65.6	67.5	66.4	54.1	64.0	67.8
dfs	65.7	67.9	65.9	54.5	64.2	68.1
ppr_w2w$_{\texttt{nf}}$	60.4	64.2	54.8	40.0	64.5	64.5
ppr$_{\texttt{nf}}$	58.6	61.3	54.9	42.2	60.9	62.9
dfs$_{\texttt{nf}}$	59.1	62.7	54.4	39.3	62.8	62.2

Table 3: Additional results on other settings of UKB. nf subscript stands for "no sense frequency". Top rows use a single sentence as context, while the bottom rows correspond to extended context (cf. Sect. 3). Best results in bold.

the latest work on this area, as well as the most frequent sense as given by WordNet counts (see Section 3). The table shows that UKB yields the best overall result. Note that Banerjee and Pedersen (2003) do not use sense frequency information.

For completeness, Table 2 reports the results of supervised systems on the same dataset, taken from the two works that use the dataset (Yuan et al., 2016; Raganato et al., 2017b). As expected, supervised systems outperform knowledge-based systems, by a small margin in some of the cases.

5 Additional results

In addition to the results of UKB using the setting in (Agirre and Soroa, 2009; Agirre et al., 2014) as specified in Section 3, we checked whether some reasonable settings would obtain better results. Table 3 shows the results when applying the three algorithms described in Section 3, both with and without sense frequencies, as well as using a single sentence for context or extended context. The table shows that the key factor is the use of sense frequencies, and systems that do not use them (those with a nf subscript) suffer a loss between 7 and 8 percentage points in F1. This would explain part of the decrease in performance reported in (Raganato et al., 2017a), as they explicitly mention that they did not activate the use of sense frequencies in UKB.

The table also shows that extending the context is mildly effective. Regarding the algorithm, the table confirms that the best method is *ppr_w2w*, followed by the subgraph approach (*dfs*) and *ppr*.

6 Conclusions

This paper presents a case where an open-source NLP software was used with suboptimal parameters by third parties. UKB was released with suboptimal default parameters, and although the accompanying papers did describe the necessary settings for good results on WSD, bad results were not prevented. The results using the settings described in the paper on newly released datasets show that UKB is the best among knowledge-based WSD algorithms.

The take-away message for open-source NLP software authors is that they should not rely on other researchers reading the papers with care, and that it is extremely important to include, with the software release, precise instructions and optimal default parameters, or better still, end-to-end scripts that download all resources, perform any necessary pre-processing and reproduce the results. UKB now includes in version 3.1 such end-to-end scripts and the appropriate default parameters.

Acknowledgements

This research was partially supported by the Spanish MINECO (TUNER TIN2015-65308-C5-1-R and MUSTER PCIN-2015-226) and the UPV/EHU (excellence research group).

References

E. Agirre and P. Edmonds. 2007. *Word Sense Disambiguation: Algorithms and Applications*, 1st edition. Springer Publishing Company, Incorporated.

E. Agirre, O. Lopez de Lacalle, and A. Soroa. 2014. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–88.

E. Agirre and A. Soroa. 2009. Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of 14th Conference of the European Chapter of the Association for Computational Linguistics*, Athens, Greece.

E. Agirre, A. Soroa, E. Alfonseca, K. Hall, J. Kravalova, and M. Pasca. 2009. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. In *Proceedings of annual meeting of the North American Chapter of the Association of Computational Linguistics (NAAC)*, Boulder, USA.

E. Agirre, A. Soroa, and M. Stevenson. 2010. Graph-based word sense disambiguation of biomedical documents. *Bioinformatics*, 26:2889–2896.

Eneko Agirre, Ander Barrena, and Aitor Soroa. 2015. Studying the wikipedia hyperlink graph for relatedness and disambiguation. In *ArXiv repository*.

Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, IJCAI'03, pages 805–810, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. 2014. An enhanced lesk word sense disambiguation algorithm through a distributional semantic model. In *COLING*, pages 1591–1600. ACL.

Francis Bond and Kyonghee Paik. 2012. A survey of wordnets and their licenses. In *GWC 2012 6th International Global Wordnet Conference*.

D.S. Chaplot and R. Sakajhutdinov. 2018. Knowledge-based Word Sense Disambiguation using Topic Models. In *AAAI*.

Nicolai Erbs, Eneko Agirre, Aitor Soroa, Ander Barrena, Ugaitz Etxebarria, Iryna Gurevych, and Torsten Zesch. 2012. Ukp-ubc entity linking at tac-kbp. In *Text Analysis Conference, Knowledge Base Population*.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. MIT Press.

Josu Goikoetxea, Eneko Agirre, and Aitor Soroa. 2015. Random walks and neural network language models on knowledge bases. In *Proceedings of the Annual Meeting of the North American chapter of the Association of Computational Linguistics (NAACL HLT 2015), pages 1434-1439. ISBN: 978-1-937284-73-2. Denver (USA).*

Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. Embeddings for word sense disambiguation: An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 897–907, Berlin, Germany. Association for Computational Linguistics.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional lstm. In *CoNLL*.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, HLT '93, pages 303–308, Stroudsburg, PA, USA. Association for Computational Linguistics.

A. Moro, A. Raganato, and R. Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association of Computational Linguistics*, 2:231–244.

R. Navigli and M. Lapata. 2010. An Experimental Study of Graph Connectivity for Unsupervised Word Sense Disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(4):678–692.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017a. Word Sense Disambiguation: A Unified Evaluation Framework and Empirical Comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110. Association for Computational Linguistics.

Alessandro Raganato, Claudio Delli Bovi, and Roberto Navigli. 2017b. Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1156–1167, Copenhagen, Denmark. Association for Computational Linguistics.

Dayu Yuan, Julian Richardson, Ryan Doherty, Colin Evans, and Eric Altendorf. 2016. Semi-supervised word sense disambiguation with neural models. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1374–1385. The COLING 2016 Organizing Committee.

Zhi Zhong and Hwee Tou Ng. 2010. It makes sense: A wide-coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations*, ACLDemos '10, pages 78–83, Stroudsburg, PA, USA. Association for Computational Linguistics.