Automatically evaluating the conceptual complexity of German texts

Freya Hewett [1 2]
[1] Humboldt Institute for Internet and Society, Berlin, Germany
freya.hewett@hiig.de

Manfred Stede [2]
[2] Applied Computational Linguistics University of Potsdam, Germany
stede@uni-potsdam.de

Abstract

Conceptual complexity is concerned with the background knowledge needed to understand concepts within a text and their implicit connections (Hulpuș et al., 2019). In the present study, a recently proposed framework from Hulpuș et al. (2019), which assesses the conceptual complexity of English newspaper articles, is replicated and adapted to German lexica entries aimed at three different age groups. The final results on the corpus of 885 German texts improve upon the original study in both a pairwise classification task and a ranking task, showing that the framework transfers well to a different language and a different genre. We release the dataset used, as well as an extended version with a total of ca. 3000 texts.

1 Introduction

Text simplification aims to reduce the complexity of a text whilst retaining the main informational content. Conceptual complexity is concerned with the background knowledge needed to understand concepts within a text, and the implicit connections between the concepts that contribute to understanding a text (Hulpuș et al., 2019). The present study aims to evaluate the conceptual complexity of German texts, by recreating a recent study from Hulpuș et al. (2019) in which they assess the conceptual complexity of English language newspaper articles from the Newsela corpus (Xu et al., 2015), which contains articles at five different levels of complexity. To do this, they develop a framework which is based on psycholinguistic theories on reading comprehension, in particular *priming*, which states that words are recognised faster if preceded by words related in meaning (Collins and Loftus, 1975).

In the present study, this framework is directly applied to German texts from three lexica designed for beginner readers, children and adults. The framework is then slightly adapted to account for nuances specific to the German language, such as compound words. The results show that the model adapts well to German texts and works well across domains. We also release the lexica dataset to foster research on German text simplification, and a script to build the dataset as the lexica grow.[1]

2 Background

The main hypothesis in Hulpuș et al.'s (2019) study is that the more priming in a text, the lower the conceptual complexity. A spreading activation (SA) framework (Quillian, 1962, 1967; Collins and Loftus, 1975) is used to illustrate the priming process. The framework compares concepts to nodes in a network, with the properties of concepts represented as labelled relational links from the node to other concept nodes. Whenever a concept is mentioned in a text, it activates other neighbouring concepts in the graph (Collins and Loftus, 1975). The amounts of activation generated by this process are used to symbolise the amount of priming in the text.

The rest of this section provides a summary of the model proposed by Hulpuș et al. (2019). The model is implemented using the DBpedia knowledge graph (Lehmann et al., 2014), which converts information from Wikipedia into a graph structure. The texts are first annotated with concepts from DBpedia using an entity linker. The SA process for each of these concepts is then calculated and consists of three functions: an input, output and activation function. Each iteration in the SA process is called a *pulse*, denoted by p. $A^{(p)}(c)$ denotes the amount of activation that node c has after pulse p. Whenever a concept is mentioned in a text, referred to as a seed concept, its activation is set to 1.0, and all other nodes are set to 0.0. At pulse 1, the SA

[1] https://doi.org/10.5281/zenodo.5196030

process is triggered and activations flows from the seed concept. The **output function** adjusts the activation according to two parameters; α is a *distance decay parameter*, which decays the activation outputted by a node at every pulse. A *firing threshold* β is also used, which limits the concepts which can fire in the next pulse. The function is defined as follows:[2]

$$A_{out}^{(p+1)}(c) = \alpha \cdot f_\beta(A^{(p)}(c)) \quad (1)$$

where $f_\beta(x) = x$ if $x \geq \beta$; 0 otherwise. The **input function** collates the activation that flows in to a target node from neighbouring (source) nodes and takes two aspects into account, the popularity and the exclusivity. The popularity is measured by how many neighbouring nodes a concept has, the exclusivity measures the semantic relatedness between two nodes by using the types of relation that connect the two nodes.[3] These two factors multiplied together are termed *accessibility*. The **input function** is defined as follows:

$$A_{in}^{(p+1)}(c) = \sum_{r \in \rho(c)} A_{out}^{(p+1)}(n_r(c)) \cdot \overline{acc}_r(c) \quad (2)$$

where $\rho(c)$ refers to the set of relations of concept c, $n_r(c)$ the neighbours of concept c through the relation r, and $\overline{acc}_r(c)$ the normalised accessibility of concept c through relation r.[4] The **activation function** computes the activation of a concept as a sum of its activation at p and its incoming activation at $p+1$:

$$A^{(p+1)}(c) = A^{(p)}(c) + A_{in}^{(p+1)}(c) \quad (3)$$

The SA process finishes when there are no more concepts which have not already fired and have an activation value higher than the β threshold. In a next step, a function (denoted as $\phi(SA(c))$) is applied to the activations that the nodes have at the last pulse of the SA process and the resulting activation scores are then subject to a forgetting process. ϕ^A uses the activation from the SA process, except for the seed concept, where the popularity score is used instead. ϕ^1 is a constant function in which all concepts which become active during the SA process receive a score of 1.

[2]Functions are taken from Hulpuș et al. (2019).

[3]The functions for popularity and exclusivity can be found in Appendix A.1 and A.3.

[4]The function for normalised accessibility can be found in Appendix A.3.

Cumulative activation (CA) calculates the SA values after they have been subject to forgetting:

$$CA^{(i)}(c) = \sum_{k=0}^{i} \gamma_{k,i} \phi(SA^{(k)}(c)) \quad (4)$$

where $CA^{(i)}(c)$ denotes the CA of a concept c at the time of reading word i. γ represents the forgetting process and is the product of three set decay factors which decrease the activation of the concepts at each encountered word, sentence and paragraph. Scores can also increase if concepts are repeated or if related concepts are mentioned later in the text. The final scores for a text are calculated at the moment the concept is encountered (AE), at the end of sentences (AEoS), paragraphs (AEoP) or the sum of all three (All). The inverse of the average of these scores is used as the conceptual complexity score for the text. The scores are used for two tasks: a pairwise classification task (i.e. which text of two texts is more conceptually complex) evaluated by calculating the percentage of pairs that are classified correctly over all the pairs in the corpus, and a ranking task (i.e. correctly ordering the texts on one topic in order of conceptual complexity) evaluated by comparing the model's ranking to the gold-standard using Kendall's tau-b, which is on a scale from -1 to +1 (Kendall, 1945).

3 Related work

Conceptual complexity. An earlier study, also by Štajner and Hulpuș (2018), on the automatic assessment of conceptual complexity uses knowledge graph based features, such as the number of neighbours a node has and the length of the shortest path connecting two nodes. They build on this work by introducing shallow and surface features based on the output of an entity linker, such as the number of unique entities in a sentence or the average distance between consecutive mentions of entities (Stajner and Hulpuș, 2020).

Feng et al. (2010) evaluate the features which best predict readability, using magazine articles designed for primary school children of different ages in a classification task. They use "discourse features" such as the density of named entities and proper nouns across a sentence or text, or the length of chains of semantic relations (such as synonym or hypernym) from an entity, based on the hypothesis that the density of named entities and proper nouns introduced in a text relates to the burden placed

on the readers' working memory and therefore the complexity level of a text.

For texts in German, Weiß and Meurers (2018) evaluate a large feature set of complexity indicators on a dataset of news subtitles and scientific articles and their counterparts aimed at children. Some of the most informative features were frequency measures calculated using different lexicons and corpora as well as content overlap within sentences. vor der Brück et al. (2008) develop a readability checker for German texts called DeLite and build so-called semantic networks for sentences, in which the word-class functions of the words and the relations between them are represented as a graph. Using 500 German texts from the municipal domain they compare human judgements on readability to automatic and conclude that indicative features include inverse concept frequency, the number of reference candidates for a pronoun and the number of propositions in a sentence.

Knowledge graphs. Knowledge graphs (KGs) have been used in a wide variety of tasks such as computing the semantic similarity of concepts (Zhu and Iglesias, 2017), finding relevant tokens in text (Bronselaer and Pasi, 2013), in recommendation systems (Joseph and Jiang, 2019) and for calculating document similarity (Paul et al., 2016). Using KGs in language-based tasks as a proxy for background knowledge is not a novel idea, and has been done in the context of argumentation mining with reasonable success (Kobbe et al., 2019; Botschen et al., 2018).

4 Data

The main data for the present study comes from a total of 885 articles from three Wiki-based lexica in German language: MiniKlexikon, Klexikon and Wikipedia. Klexikon is aimed specifically at children aged between 6 and 12 (Dunemann, 2016) and MiniKlexikon is designed for children who are beginner readers, and is therefore an even simpler version of the Klexikon. We make the assumption that the three different sub-corpora represent three different levels of conceptual complexity due to the target groups they are written for: younger children, children and adults. Children have less prior knowledge so therefore a text written for them should require less background knowledge; this aspect is explicitly mentioned in the guidelines for writing articles for the MiniKlexikon.[5] As Wikipedia articles can be extremely long, in comparison to the other two lexica, only the introduction or abstract was taken for the purposes of the current study. Any Klexikon articles longer than 2800 characters were excluded, as well as any articles where parallel topics did not exist across all sub-corpora. This resulted in 295 texts for each level. The different sub-corpora will be referred to hereafter as level 0 (MiniKlexikon), level 1 (Klexikon) and level 2 (Wikipedia). Table 1 shows that the level 1 sub-corpus contains the longest articles, but the average sentence length gets longer as the complexity level increases. Examples from the corpus can be seen in Table 2.

Sub-corpus	Texts	Avg. AL	Avg. SL
Level 0	295	134.86	9.57
Level 1	295	305.45	13.29
Level 2	295	169.89	18.41

Table 1: Average length of articles (AL) and average sentence length (SL) in the three sub-corpora (tokens).

5 Experiments

The system from Hulpuș et al. (2019) was first replicated, adapted only by changing the language of the DBpedia graph to German. As in the original study, different parameters were experimented with: the extent of the forgetting process, γ, – the so-called type of decay – and the ϕ function, which is the function applied to the values which result from the SA process. The distance decay parameter α and the firing threshold β, two parameters which control the amount of nodes activated in each SA step, were not experimented with and the best performing values from the original study were used, 0.25 and 0.01 respectively. The system was then applied to all 885 texts in the lexica corpus. The results can be seen in Table 3: the average accuracy for pairwise classification using the best parameters from the original study (as documented in (Štajner et al., 2020)) was .86, which is the same as the original system for English texts. The best parameters for the German texts – as can be seen in the right-hand side of Table 3 – increased the average accuracy for the pairwise classification to .89. In both cases the AEoS score provided the best results.

[5] https://miniklexikon.zum.de/index.php?title=Hilfe:Regeln&oldid=23440

Level 2	Simplified (level 0/1)	Simplification
The name Allosaurus **is derived from the Greek language and** translates to 'different lizard'.	The name Allosaurus means something like 'different lizard'.	removal of non-essential concepts that demand more background knowledge
Amsterdam is the capital city and the **most populous** city in the **Kingdom of** the Netherlands.	Amsterdam is the capital city of the Netherlands. Amsterdam is also the **biggest** city in the Netherlands.	replacement of non-essential demanding concepts with more commonly known ones
Furthermore, **astronomy strives to understand** the universe as a whole, its origins and its development.	Astronomers investigate how space originated.	avoidance of abstract concepts

Table 2: Translated examples of conceptual simplification from the lexica corpus created for the present study. The types of simplification are taken from (Štajner and Hulpuș, 2018).

decay	medium decay, ϕ^1				strong decay, ϕ^A			
score	**AE**	**AEoS**	**AEoP**	**All**	**AE**	**AEoS**	**AEoP**	**All**
0-1	.56	**.93**	.89	.92	.58	.87	.82	.88
0-2	.35	.88	.69	.79	.52	**.94**	.82	.91
1-2	.30	.76	.48	.59	.48	**.87**	.62	.76

Table 3: The accuracy scores for the pairwise classification task with the parameters from the original study (Hulpuș et al., 2019). The scores on the left use the best parameters for the Newsela corpus, the scores on the right use the best parameters for the lexica corpus. The highest accuracy for each pair of levels is highlighted in bold.

5.1 Adaptations

Manual inspection of the concepts annotated by the entity linker, DBpedia Spotlight (Mendes et al., 2011), revealed some inaccurate annotations, particularly at a confidence level of 0.35, which is the level used by Štajner et al. (2020). Nouns with capitalised articles are often tagged as films or bands that go by the same name such as *the depth* (*Die Tiefe*). We experimented with different confidence levels (0.35 to 0.65, at intervals of 0.05) and with an alternative entity linker for German, TagMe, with the same amount of the equivalent confidence levels (Ferragina and Scaiella, 2010, 2012). Whilst the accuracy of the tagged concepts did appear to improve, neither the confidence values nor the TagMe entity linker improved the scores for either task. Another approach was taken to try and improve the accuracy of the entity linker for the specific task of solely tagging concepts. In the context of the present model, a concept is simply defined, by proxy, as a node in the DBpedia KG. By analysing the texts in the corpus, this definition could be elaborated upon to say that concepts are nodes in the DBpedia KG that are also nouns, verbs, adjectives, adverbs or cardinal numbers. The whole corpus was tagged with Part-of-Speech tags using TreeTagger (Schmid, 1999) and entity annotations were removed that did not fit this definition. This reduced the amount of concepts tagged by approximately 15%.

Another challenge that the entity linkers have to deal with, that is somewhat unique to the German language, is the high presence of compound words such as *Pumporgan*: literally pump organ, "heart". *Pumporgan* does not have its own DBpedia page which implies it is a somewhat novel compound. Most novel compounds are transparent, as it can be assumed that the reader is seeing them for the first time, so they have to be able to be understood by the context and the meaning of the constituents (Smolka and Libben, 2017). In this way, annotating *Pumporgan* with the individual concepts *Pump* and *Organ* would reflect the process that a reader goes through when processing a novel compound, and would be the ideal behaviour for the entity linker. To facilitate the tagging of such compounds, a compound splitter (Ziering and van der Plas, 2016) was applied to the level 2 data before the entity linking stage. According to the MiniKlexikon guidelines[6], unusual compounds should be hyphenated and so the splitter was not used on levels 0 and 1, and instead hyphenated words were separated.

We also experimented with different ϕ functions. ϕ^U refers to *unchanged*, so taking the SA scores as is, ϕ^{red} refers to *reduced* so simply applying the forgetting process to the entity linker output, leaving out the SA process completely and ϕ^{pop} refers to *popularity*, and also leaves out the SA process whilst including the popularity scores of the tagged concepts. The equations for these ϕ functions can be found in Appendix A.2. We also introduced an **AEoD** score which sums up the score for the whole document, and tried out different combinations of calculating the **All** score.

[6] https://miniklexikon.zum.de/index.php?title=Hilfe:Regeln&oldid=20790

System	ϕ	Decay	AA	tau-b
original framework, English texts	ϕ^1	medium	.86$^+$	.82*
framework replication, German texts	ϕ^A	strong	.89	.79
PoS based outlier removal	ϕ^A	strong	.89	.78
compound splitting, just level 2	ϕ^A	strong	.89	.79
compound splitting, all levels	ϕ^A	strong	.70	.41
AEoD score	ϕ^A	strong	.52	.04
All + AEoD	ϕ^A	strong	.81	.62
AEoS + AEoP	ϕ^A	strong	.87	.74
unchanged scores	ϕ^U	medium	**.91**	**.83**
entity linker + forgetting	ϕ^{red}	medium	**.91**	**.83**
entity linker + popularity + forgetting	ϕ^{pop}	strong	.85	.70

Table 4: The average accuracy (AA) for the pairwise classification task and tau-b for the ranking task using the AEoS scores for various models, with different ϕ and decay parameters (only the best-performing combinations for each system are shown). +From Štajner et al. (2020). *From Hulpuș et al. (2019): the tau-b results are calculated using an entity linker which is not publicly available; a direct comparison is therefore not possible.

5.2 Results & discussion

The results on the lexica corpus can be seen in Table 4. The best accuracy and tau-b score is for the model with unchanged scores from the SA process (ϕ^U) and the model which just uses the seed concepts and a forgetting process (ϕ^{red}). This second model, ϕ^{red}, also has the advantage of being much more efficient than the models which involve the spreading activation process. This is an improvement of 5 percentage points on the original study, although it is worth mentioning that the results can not be directly compared due to the different nature of the datasets. The lexica corpora used in this study are on 3 different levels (as apposed to the Newsela corpus which has 5 levels) and the texts do not necessarily represent parallel translations.

As can be seen in Table 1, the average sentence lengths of the different levels of the corpus increase as the complexity increases. In fact, using average sentence length as a sole feature for the ranking task results in a tau-b score of .87. However, for downstream tasks such as automatic simplification or summarisation, a content based classification of complexity – such as the conceptual complexity value – could prove to be a lot more informative. Another use case for conceptual complexity is for texts that may not conform to this pattern of shorter sentences for less complexity. For example, when simplifying complex sentences by including examples or extra clauses that explain difficult terms, the sentence length will increase as the complexity level decreases.

As the success of a framework that uses a specific KG as a proxy for long-term memory is obviously highly dependent on the quality of the KG, a manual inspection of the DBpedia KG was carried out. This showed that nodes are not always linked to each other in an intuitive way, with many nodes completely isolated. A random sample of 30 results from the popularity function showed that the node *multiplication* scores 0, as it has no neighbours, and *Helgoland* and *Calligra Suite* score higher than *ruler* or *hair*, which may not correspond to an average reader's level of familiarity. Working with a different KG or calculating the popularity or familiarity of concepts in an ontology-independent way could yield more accurate results; we leave this to future work.

6 Conclusion & outlook

In this study, the conceptual complexity of German lexicon entries was examined by replicating and adapting a spreading activation framework proposed by Hulpuș et al. (2019). When compared to the results from the study using the same entity linker (Štajner et al., 2020), the current implementation improves the average accuracy score for pairwise classification by 5 percentage points. This shows that the adapted framework also works with shorter texts and can be adapted to work with languages other than English. We release the main dataset used and a script to continually update it. An interesting direction for future research would be a closer examination of the way concepts are connected on a text level, implicitly and explicitly, and how the discourse structure affects complexity.

Acknowledgments

Thank you to the anonymous reviewers for their very helpful comments. This research is funded by the German Federal Ministry of Education and Research (BMBF).

References

Teresa Botschen, Daniil Sorokin, and Iryna Gurevych. 2018. Frame- and entity-based knowledge for common-sense argumentative reasoning. In *Proceedings of the 5th Workshop on Argument Mining*, pages 90–96. Association for Computational Linguistics.

Antoon Bronselaer and Gabriella Pasi. 2013. An approach to graph-based analysis of textual documents. In *8th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2013)*, pages 634–641. Atlantis Press.

Tim vor der Brück, Sven Hartrumpf, and Hermann Helbig. 2008. A Readability Checker with Supervised Learning using Deep Syntactic and Semantic Indicators. *Informatica*, 32:429–435.

Allan Collins and Elizabeth Loftus. 1975. A Spreading Activation Theory of Semantic Processing. *Psychological Review*, 82:407–428.

Tabea Dunemann. 2016. Ins Netz gegangen: Klexikon.de. Wenn Wissen mitmachen lässt. *tv diskurs*, 76:112–113.

Lijun Feng, Martin Jansche, Matt Huenerfauth, and Noémie Elhadad. 2010. A Comparison of Features for Automatic Readability Assessment. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 276–284. Association for Computational Linguistics.

Paolo Ferragina and Ugo Scaiella. 2010. TAGME: On-the-Fly Annotation of Short Text Fragments (by Wikipedia Entities). In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1625–1628. Association for Computing Machinery.

Paolo Ferragina and Ugo Scaiella. 2012. Fast and Accurate Annotation of Short Texts with Wikipedia Pages. *IEEE Software*, 29(1):70–75.

Ioana Hulpuş, Narumol Prangnawarat, and Conor Hayes. 2015. Path-Based Semantic Relatedness on Linked Data and Its Use to Word and Entity Disambiguation. In *The Semantic Web - ISWC 2015*, pages 442–457, Cham. Springer International Publishing.

Ioana Hulpuș, Sanja Štajner, and Heiner Stuckenschmidt. 2019. A Spreading Activation Framework for Tracking Conceptual Complexity of Texts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3878–3887. Association for Computational Linguistics.

Kevin Joseph and Hui Jiang. 2019. Content based News Recommendation via Shortest Entity Distance over Knowledge Graphs. In *Companion of The 2019 World Wide Web Conference, WWW 2019*, pages 690–699. ACM.

Maurice G. Kendall. 1945. The treatment of ties in ranking problems. *Biometrika*, 33:239–251.

Jonathan Kobbe, Juri Opitz, Maria Becker, Ioana Hulpus, Heiner Stuckenschmidt, and Anette Frank. 2019. Exploiting background knowledge for argumentative relation classification. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*, volume 70, pages 8:1–8:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, and Christian Bizer. 2014. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 6:1–29.

Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. DBpedia Spotlight: Shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8. Association for Computing Machinery.

Christian Paul, Achim Rettinger, Aditya Mogadala, Craig Knoblock, and Pedro Szekely. 2016. Efficient Graph-Based Document Similarity. In *The Semantic Web. Latest Advances and New Domains*, pages 334–349. Springer, Cham.

Ross Quillian. 1962. A revised design for an understanding machine. *Mechanical Translation*, 7:17–29.

Ross Quillian. 1967. Word concepts: A theory and simulation of some basic semantic capabilities. *Behavioral Science*, 12:410–430.

Helmut Schmid. 1999. *Improvements in Part-of-Speech Tagging with an Application to German*, pages 13–25. Springer Netherlands.

Eva Smolka and Gary Libben. 2017. 'Can you wash off the hogwash?' : semantic transparency of first and second constituents in the processing of German compounds. *Language, Cognition and Neuroscience*, 32(4):514–531.

Sanja Štajner and Ioana Hulpuș. 2018. Automatic Assessment of Conceptual Text Complexity Using Knowledge Graphs. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 318–330. Association for Computational Linguistics.

Sanja Stajner and Ioana Hulpuș. 2020. When shallow is good enough: Automatic assessment of conceptual text complexity using shallow semantic features. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1414–1422, Marseille, France. European Language Resources Association.

Sanja Štajner, Sergiu Nisioi, and Ioana Hulpuș. 2020. CoCo: A Tool for Automatically Assessing Conceptual Complexity of Texts. In *Proceedings of*

The 12th Language Resources and Evaluation Conference, pages 7179–7186. European Language Resources Association.

Zarah Weiß and Detmar Meurers. 2018. Modeling the Readability of German Targeting Adults and Children: An empirically broad analysis and its cross-corpus validation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 303–317. Association for Computational Linguistics.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in Current Text Simplification Research: New Data Can Help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

Ganggao Zhu and Carlos A. Iglesias. 2017. Computing semantic similarity of concepts in knowledge graphs. *IEEE Transactions on Knowledge and Data Engineering*, 29(1):72–85.

Patrick Ziering and Lonneke van der Plas. 2016. Towards Unsupervised and Language-independent Compound Splitting using Inflectional Morphological Transformations. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–653. Association for Computational Linguistics.

A Appendices

A.1 Popularity function

The popularity function is defined as follows:

$$pop(c) = \frac{log(D(c))}{log(|V| - 1)} \tag{5}$$

where $D(c)$ denotes the number of neighbours of concept c, and $|V|$ the total number of concepts in the KG.

A.2 ϕ functions

Functions for ϕ^A and ϕ^1 as described in Section 2, taken from Hulpuș et al. (2019):

$$\phi^A(SA(c)) = \begin{cases} SA(c), & \text{if } SA(c) < 1.0 \\ pop(c), & \text{if } SA(c) \geq 1.0 \end{cases} \tag{6}$$

$$\phi^1(SA(c)) = 1 \quad \text{if} \quad SA(c) > 0.0 \tag{7}$$

Additional functions for ϕ^U, ϕ^{red} and ϕ^{pop}:
ϕ^U, which refers to *unchanged* and simply takes the values as-is from the SA process and is defined as follows:

$$\phi^U(SA(c)) = SA(c) \tag{8}$$

ϕ^{red}, which refers to *reduced*, which just takes the seed concepts and applies forgetting, and is defined as follows:

$$\phi^{red}(SA(c)) = \begin{cases} 0.0, & \text{if } SA(c) < 1.0 \\ SA(c), & \text{if } SA(c) \geq 1.0 \end{cases} \tag{9}$$

ϕ^{pop}, which refers to *popularity*, which just calculates the popularity for activated concepts and applies forgetting, which is defined as follows:

$$\phi^{pop}(SA(c)) = pop(c) \quad \text{if} \quad SA(c) > 0.0 \tag{10}$$

A.3 Differences to original study (Hulpuș et al., 2019)

Our replicated framework was tested with a subsample of 25 Newsela texts (Xu et al., 2015). Using the original rankings as published here[7] as gold standard, our replicated system had a tau-b of .9.

The reasons for this slight difference could be due to the following reasons: Štajner et al. (2020) use a different exclusivity calculation (cf. 12), the Newsela texts used for the present study are formatted slightly differently and do not have paragraph information, two equations (11, 6) were adjusted as the original equations in (Hulpuș et al., 2019) do not fully match the descriptions in the accompanying paper. In addition to this, Štajner et al. (2020) do not specify if they use a support parameter when using the entity linker DBpedia Spotlight. This slightly limits the pool of neighbouring nodes which is returned. In the present study we use a support value of 20.

The normalised accessibility function:

$$\overline{acc}_r(c) = \frac{acc_r(c)}{(acc_r(c) + \sum_{r' \in \rho(n_r(c))} acc_{r'}(n_{r'} \circ n_r(c)))} \tag{11}$$

The exact equation for exclusivity was not listed in the paper, and at the time of replicating the framework, no further information was available. The following function was used, adapted from the function in (Hulpuş et al., 2015):

$$excl(r) = \frac{1}{|* \xrightarrow{\tau} x \xrightarrow{\tau} *| + |* \xrightarrow{\tau} y \xrightarrow{\tau} *| - 1} \tag{12}$$

[7] https://github.com/ioanahulpus/cocospa/blob/master/results/newsela.csv