A Pseudonymization Prototype for Hungarian

Attila Novák ✉
Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary

Borbála Novák ✉
Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Budapest, Hungary

Abstract

In this paper, we present a pseudonymization prototype for Hungarian, an agglutinating language with complex morphology, implemented as a web service. The service provides the following functions: entity identification and extraction; automatic generation and selection of replacement candidates; automatic and consistent replacement and reinflection of entities in the final pseudonymized document. The named entity recognition model applied handles names of persons well, and it has decent performance on other entity types as well. However ID-like entities need to be handled separately to achieve proper performance (not handled in the current prototype version). For automatic replacement candidate generation, a simple entity embedding model is used. We discuss the performance and limitations of the prototype in detail.

2012 ACM Subject Classification Computing methodologies → Natural language processing

Keywords and phrases named entity recognition, morphological reinflection, pseudonymization, entity embedding model

Digital Object Identifier 10.4230/OASIcs.SLATE.2023.3

Funding The research presented in this paper was implemented with support provided by grant FK 125217 of the National Research, Development and Innovation Office of Hungary financed under the FK 17 funding scheme.

1 Introduction

Machine learning-based NLP models are often domain dependent to some extent in the sense that their performance often depends on how similar the features (topic, style, vocabulary) of the text to which the model is applied are to the features of the texts used to train the model. The performance of models can often be significantly improved if (ideally a significant amount of) in-domain training data is available.

However, access to types of texts containing sensitive personal data (such as medical or crime-related information) is severely restricted, and this can be a serious obstacle to the development of high-quality models for handling texts in such domains. Consistent automatic replacement of personal data in texts with similar but fictitious data is a possible solution to this problem. The type of solution of which we outline a working research prototype in this paper could provide a general solution to the legal problems that hinder the publication and use of texts containing sensitive data, and thus contribute significantly to the development of high-quality language models in these domains.

The objective of the research presented here was to develop a prototype that uses a high-precision and high-coverage named entity recognition algorithm to identify names and other personal data together with their entity type in text, and then associates fictitious but natural-looking names and data with them, replacing the occurrences of names and data (including suffixed forms) in the text in a consistent way. This way, we obtain texts that no longer contain real personal data and are therefore no longer constrained by the restrictions pertaining to the original data. These can thus be made available for training or fine-tuning

© Attila Novák and Borbála Novák;
licensed under Creative Commons License CC-BY 4.0

12th Symposium on Languages, Applications and Technologies (SLATE 2023).
Editors: Alberto Simões, Mario Marcelo Berón, and Filipe Portela; Article No. 3; pp. 3:1–3:10
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

language models that can handle texts in the given domain. This type of models can also be used to replace the identified entities and other data with unique identifiers to provide proper data masking. Typically, pseudonymization means that a given code key can be used to recover the original information, and, indeed, the systematic recoverability of information is often a desired objective. When unique identifiers are used to replace the original names and data, these are recognizably different from real data. However, to achieve our original objective (i.e. to obtain a restriction-free version of a corpus of texts in a domain typically restricted by the presence of sensitive information), we need a solution that results in an output that is essentially indistinguishable from the original in terms of lexical distribution, thus we need to replace names with false names of similar distribution rather than with identifiers. This solution is usually referred to as data masking. For this purpose, the code key that could be used to recover the original data is unnecessary, and it can and should be discarded or at least is to be kept separately and subject to technical and organizational measures to ensure non-attribution to any identified or identifiable person.

Reversal of pseudonymization may be possible in some edge cases using externally available information even in the absence of the code key, if unmodified information in the context can be used to infer the identity of the person in an unambiguous manner. For this not to be the case, data masking in general needs to be applied not only to names of persons but also to other entities like institutions, places, facilities etc. in order to mask the contexts as well.

What makes the proper data masking process non-trivial in the case of morphologically complex languages like Hungarian is that nouns may have dozens-to-hundreds of possible inflected and derived forms that all should be recognized, disambiguated and consistently replaced and reinflected in context.

2 Method

The prototype system we present in this paper contains the following components (for detailed discussion of components and functions see the subsections below):

1. A named entity recognition (NER) model to identify the entities to be replaced and the type of each entity.
2. A model for disambiguated morphological analysis and lemmatization to identify the lexical form of entities to be replaced and the actual morphological form the name to be used for substitution must take in the given context.
3. A simple extraction module that compiles a collection of extracted entities (in a normalized/lemmatized form) along with their types and frequencies.
4. A model that can suggest suitable replacement candidates for the identified entities. Ideally, replacements should be done in a manner that is consistent throughout the document.
5. A model that replaces the identified entity names and reinflects them to match the grammatical context. For this, a morphological generator model is used.
6. The prototype is implemented as a dockerized web service taking json input and returning json output.

The web service call API has the following functions:

1. Function *spans* to preprocess text and extract entities (these are added to the input json).
2. Function *suggest* to suggest replacements (entities are enriched with suggestions and replacements) creating a replacements configuration. If the input does not contain preprocessed data and extracted entities, function *spans* is performed before suggesting replacements.

3. Function *replace* to automatically replace entities in text based on the replacements configuration reinflecting them in context. *spans* and/or *suggest* are also performed, if needed. The replacements configuration can be modified manually or in an automatic manner by an external process before calling *replace*.

2.1 The named entity recognition (NER) model

The NER model we used is based on the Hungarian named entity corpus *NerKor+Cars-OntoNotes++* [3], a 1.04M-token corpus covering 85 thousand entities of a relatively fine-grained 28-class entity type set. The entity classes include the 18 entity types covered by the *Ontonotes 5* corpus [8], and further types (such as *media, social media, awards, motor vehicles, projects*) differentiated when creating the corpus. We used this corpus to finetune a transformer encoder model to perform token classification. It is based on the Hugging Face Transformers tool set, and it is available at the Hugging Face Hub[1].

The key entity types targeted in the current prototype application are: *persons* (PER), *organizations* (ORG), *geopolitical entities* (GPE): i.e. names of settlements, countries and geopolitical regions like counties, and *facilities* (FAC), which include streets/roads and other public spaces. The latter two types are parts of addresses, an important target data type for the data masking task. The *NerKor+Cars-OntoNotes++* annotation does not cover nested entities, thus addresses are not annotated as a whole, only their parts are identified by the named entity annotation model. Nevertheless, given the unnested annotation, since the consistent replacement of names (including settlement names) in the document is desirable, annotation of addresses as a sequence of settlement and street address is not problematic. The *NerKor+Cars-OntoNotes++* corpus (in contrast to earlier NER datasets) also identifies derived forms of names.

The model also identifies *dates, times, time durations, numerical values, quantities, amounts of money* and *certain types of ID's* in addition to named entities. These entities are not only annotated but are also extracted. While the replacement of these types of entities is necessary in a full-fledged data masking annotation solution, the current prototype, while it identifies and annotates most of these, it does not automatically suggest a replacement for them (while it does generate automatic replacement candidates for named entities).

2.2 A morphological analysis and lemmatization

The models used for disambiguated morphological annotation and lemmatization are the *emMorph* morphological analyzer [4, 5] integrated with the *PurePos/emTag* tagger [6]. We used the e-magyar/emtsv pipeline [7] to integrate the morphological analyzer, the tagger and the named entity recognizer. However, we used improved versions of all tools instead of the ones originally shipped with e-magyar.

The most important improvement concerns the NER model: the model in e-magyar distinguishes only four entity types: persons, organizations, locations and miscellaneous (all other types of named entities), and it was trained on a much smaller 226k-token corpus in a limited domain (business news). Thus the original NER model does not properly identify or differentiate some entity types important in a data masking application (e.g. geopolitical entities, facilities and other geographical locations are not differentiated, and no derived

[1] `https://huggingface.co/novakat/nerkor-cars-onpp-hubert`

forms of named entities are recognized).[2] In addition, it has suboptimal performance on generic (non-business-domain) texts due to its training data being limited to a single domain: the BERT-based model originally featured in e-magyar has an F_1 score of only 0.8439 on the union of test sets of the business news NER corpus and NerKor corpus, while for the model trained on NerKor (using the same architecture), F_1 was found to be 0.9197 on the same joint test set.

We have also retrained the tagger model on an improved version of its original training corpus where annotation errors due to earlier erroneous conversion of morphosyntactic annotation were fixed. We also extended the stem database of the morphological analyzer to improve coverage of named entities both in analysis and generation.

2.3 Marking and extraction of entities

All entities identified by the NER model are marked in the text to be processed, they are normalized to a lexical form based on the lemmatization provided by the morphological analyzer, and the normalized form of the entities is extracted as a json dictionary including frequency and entity type data. Normalization affects the rightmost element of multi-word entities, as case marking and other inflections are attached to the head of the noun phrase, which is on the right in Hungarian. Normalization is needed to create a single representative lexical form for all inflected forms of each entity to make the consistent replacement and reinflection of the entities possible. The entity annotation including morphological analysis on heads of noun phrases is added to the original text as markup. This representation is used later to replace and reinflect entities.

2.4 The model for replacement suggestion

The dictionary of extracted entities serves as a basis for the configuration of the replacements to be performed. Replacement candidates are automatically added to this data structure when the *suggest* function of the web service call API is called.

In the current prototype, we used a simple static word/entity embedding model to automatically populate the replacements configuration candidates section. The embedding model was trained using an annotated 2-billion-word web-crawled corpus. The annotation followed the format presented in [2]: inflected words are represented by a sequence of two tokens: one consisting of the lemma and the PoS tag, and another independent token representing the morphological endings. The following example shows the representation of the sentence *Szeretlek, kedvesem.* ‘I love you, my darling.’:

```
szeret[/V] [Prs.1Sg>2] ,[Punct] kedves[/N] [Poss.1Sg]
love       [I, you]    ,        darling    [my]
```

This representation, while no information is lost, improves the quality of the word embedding model compared to one created from surface word forms in several ways: by assigning a separate representation to lexical items of different part of speech, by effectively reducing data sparseness problems following from the great variety of rare inflected word forms, and thus by improving the representation of lemmata.

[2] In Hungarian, derivational suffixes are used to derive adjectives from e.g. names of locations. These are not capitalized: *budapesti* ‘of/in/to Budapest’

This scheme was extended to include a representation of entities. In addition to morphological annotation, the corpus was also annotated using the NER model presented in Subsection 2.1. Lemmata of heads of entities were also annotated by the entity type in addition to PoS. Sentences containing multi-word entities were represented twice in the training data: once with the whole entity represented by a single token (with underscores between words of the phrase) and once by each non-terminal surface word form of each multi-word entity appearing as an independent token. This made it possible to generate independent representations for the whole entity and its constituent parts.

```
Iványi_Márta[/N]=[PER] [Nom] -[Punct] szoprán[/Adj] [Nom]
Iványi Márta[/N]=[PER] [Nom] -[Punct] szoprán[/Adj] [Nom]
Márta Iványi                 -        soprano
```

This feature of the embedding model makes it possible for the data masking tool to handle different entity types in a different manner. E.g. replacement candidates for names of persons are generated by handling surnames and given names independently. This ensures that e.g. kinship relations among persons mentioned in the text reflected by identical surnames are preserved by handling the surnames consistently. In contrast, facilities (e.g. street names) or the names of organizations are treated by the algorithm as one unit, thus it does not make up completely fictitious street or organization names.

The embedding model was trained using the fastText CBOW algorithm [1]. For the entity types handled by the automatic replacement model of the prototype, a random shuffled sample (currently 5 items) of the top (currently 50) nearest neighbors according to cosine similarity from the embedding model are added as replacement candidates to the replacements configuration with the second candidate selected (the first candidate is always the original entity itself). For quantities, date and ID entities, currently no additional replacement candidate is generated (see limitations of the current model below in Section 3). The replacement candidates generated for the female name *Bulcsu Mariann* are shown as an example below (with the candidate *Putz Evelin* selected).

```
"Bulcsu Mariann[/N]": {
  "allsugg": [["Bulcsu","Putz","Gutbrod","Maczucza","Südy","Gálos"],
     ["Mariann","Evelin","Zita","Judit","Krisztina","Erika"]],
  "frq": 1,
  "repl": "Putz Evelin",
  "norm": "Bulcsu Mariann",
  "type": {"PER": 1}
},
```

2.5 Replacement and re-inflection of entities

Entities in the annotated original text are replaced based on the replacements configuration passed to the *replace* API function. The normalized form of entities as marked in the annotation is first replaced by the normalized form of the selected replacement candidate, and then the latter is re-inflected using a morphological generator model created from the morphological analyzer. The morphological generator uses the lemma of the head and the disambiguated morphosyntactic analysis in the original annotation to generate the contextually appropriate inflected form of the replaced entity.

3 Limitations

The current prototype implementation has a number of limitations.

It does not automatically replace numeric, identifier or date-type entities, nor does it recognize all of these types (especially some types of identifiers such as vehicle registration numbers or telephone numbers). The latter is a major limitation from a GDPR point of view, which can be handled by either creating extra training data for the underlying NER model and retraining it, or by applying an independent identifier recognition model. Automatic replacement of numerical, date-type etc. entities identified by the current NER model can be simply solved by implementing and calling a module that generates and injects a replacement for entities of these types in the entity replacements configuration before invoking the replacement function. Most of these identifiers are easily detectable using simple regex-based patterns. We did not deal with this issue in the current prototype.

The system does not currently detect and consistently handle informal references to entities or accidental misspellings or other name variations. E.g. nicknames referring to persons in the text are not identified, and the suggested replacement for them is not consistent with the the replacement of the full name (e.g. the suggested replacement for a name like *Johnny* may be e.g. *Frankie* while the replacement candidate for *John* may be e.g. *Michael*).

The automatic replacement of names referring to geographic entities are not currently handled consistently either. E.g. while the original text may mention settlements which are close to each other geographically and also the name of the county[3] where all of them are located may be mentioned, the automatically selected replacements may not have the same properties.

Addresses are not handled completely consistently, either. The zip code may be inconsistent with the rest of the address, and the substituted settlement may not have the independently replaced street name. These inconsistencies in the output may negatively impact the quality of a complex language model trained on the output of the data masking tool, thus a better model for the replacement of place names and addresses would be desirable in a follow-up model version. It is a question to what extent it is desirable to mask settlement names. It is possible that masking street addresses provides satisfactory data masking to prevent possible data leaks.

In addition, some personal data may “leak” through the system due to imperfect recall of the named entity recognizer. However, the subsequent identification of such entities in the text is only possible in certain presumably rare cases: if almost all occurrences of a name are replaced except one, the inconsistent contextually unanchored single occurrence of the name may be inferred to be the original name.

Note that for some linguistic tasks requiring domain adaptation, consistent replacement of names is not necessarily required: a solution where names in the document are replaced randomly may be sufficient. While the resulting text is not suitable for training e.g. of models for co-reference resolution, it is certainly not a problem in this case if some data has not been replaced by chance, because there is no way of deducing from the text which elements of the original text remained unchanged.

A further problem may be the (lack of) handling of nested entities (when one name element contains another name element). The frequency of such errors can be estimated by measuring them on test data, and their actual impact can be assessed by manual inspection of automatically pseudonymized example texts.

[3] A county is a large administrative area in Hungary: Hungary consists of 19 counties.

4 Evaluation

We performed evaluation of the prototype on a 14.5k-word sample of police witness interview reports, which had been manually pseudonymized before. Manual pseudonymization was performed at the data processing company of the Ministry of the Interior, thus we were provided test data that had already been manually made GDPR-compliant. Nevertheless, the data was not made publicly available. The training data for the NER model did not contain any data similar to this genre.

Table 1 Performance of the NER model on the test corpus. Scores were reported by the CONLL-2003 NER evaluation script.

Frq. Ratio/Acc	Type	P	R	F_1
99.48%		94.74	93.33	94.03
37.59%	PER	99.01	99.01	99.01
14.26%	CARDINAL	97.44	98.70	98.06
12.96%	GPE	96.88	88.57	92.54
6.85%	FAC	94.59	94.59	94.59
6.67%	DATE	87.80	100.00	93.51
4.07%	ORG	86.67	59.09	70.27
3.89%	ORDINAL	100.00	100.00	100.00
3.70%	TIME	95.24	100.00	97.56
2.41%	DUR	100.00	100.00	100.00
1.67%	ID	100.00	55.56	71.43
1.30%	QUANTITY	100.00	100.00	100.00
1.11%	MONEY	100.00	100.00	100.00
1.11%	TEL	0.00	0.00	0.00
1.11%	CAR	100.00	33.33	50.00
0.93%	LOC	35.71	100.00	52.63
0.37%	AGE	100.00	100.00	100.00
0.00%	PROD	0.00	0.00	0.00

Table 1 shows the entity recognition performance on this test corpus. The overall F_1 score was 0.943, which is quite acceptable (only exact token span match is rewarded). The entity types in the table are ordered by their frequency in the gold test data. The most common entities are *person* names, for which we get an almost perfect recognition performance. This is quite reassuring, as this is the most important entity type from a GDPR point of view. As for now, *cardinals* also subsume zip codes and house numbers, as these have not been distinguished in the original NER model. But they are easily identifiable given their distribution relative to other parts of addresses. The relatively lower recall for geopolitical entities is caused by a) some informal references to settlement names (where instead of the official name of a settlement, a colloquial form was used) that were mostly mistagged by the model as LOC (this is a minor issue, its impact on replacement is that the suggested replacement for the name was of the wrong semantic category), and b) by the model sometimes missing some derived forms of settlement names. The relatively low recall for ORG entities is due to references in the texts to specific police headquarters and their departments in all caps, which was sometimes left untagged or tagged as LOC (except for the settlement name within the name, which is identified as GPE). This is not a grave problem either with regard to the performance of the data masking application. The model cannot identify phone numbers and some types of ID's, as mentioned in Section 3 on limitations. Other errors included the model tagging some car occurrences in the corpus as products rather than assigning the more specific CAR tag.

We also evaluated replacement and reinflection performance. The test set contained 894 identified entities, of which 474 were replaced. The main error types are shown in Table 2. The second column shows the ratio of errors in the replacement configurations, the third column in the actual occurrences in the test corpus.

The majority of unreplaced entity occurrences was due to our decision not to change them in the current prototype (dates, times, quantities, ID's, etc.). 44 entities (8.5% of the 518 entities that should have been replaced) remained unreplaced due to some error in entity identification or automatic suggestion generation. This ratio definitely needs to be improved in a production version (e.g. by replacing/improving the static-embedding-based suggestion algorithm). The inconsistent replacement of nicknames mentioned in Section 3 affects 2.7% of entities to be replaced. The replacement was deemed 'improper' in 5% of the cases: the selected replacement is of a different name type than the original, or a foreign first name was suggested as a replacement of a Hungarian name.[4] Adding a simple filtering mechanism to the handling of persons names could alleviate these problems. In many cases, misclassification of entity type by the NER model could be identified as the cause of improper replacement. Another 0.6% was replaced inconsistently due to some other factor (e.g. nonstandard usage of street address).

Table 2 Error types and actual occurrences in entity replacement due to an error in the automatic entity detection or replacement candidate generation/selection.

error type	entries in the configurations	affected occurrences
all	40 (19.7)	87 (16.8)
unreplaced	23 (11.3)	44 (8.5)
improper	9 (4.4)	26 (5.0)
nickname	5 (2.5)	14 (2.7)
inconsistent	3 (1.5)	3 (0.6)

Reinflection errors in the final document can be traced back either to morphological analysis errors (often due to some spelling error, affecting 1.8% of replaced entities) or affect locative cases of settlement names (0.8%). In Hungarian, names of geopolitical entities and public places (like street names) take a locative form either in the superessive *(on)* (e.g. *Budapest, Magyarország* 'Hungary'), or in the inessive *(in)* case (e.g. *Madrid, Spanyolország* 'Spain'). This is a lexical property of the name, and replacement of a name with another that belongs to the other group results in improper case inflection. This problem will need to be addressed in a future model update.

5 In the context of ChatGPT

The prototype we presented in this paper is based on a traditional NLP pipeline (although some elements of the pipeline have a neural implementation). Although it was created before ChatGPT's earthquake-like debut, we felt compelled to check whether we can get ChatGPT perform the same task out of the box. However, we did not manage to prompt ChatGPT

[4] This is due to the fact that a specific subset of foreign first names (of celebrities) are popular among members of some specific social groups, while other similar foreign first names are not.

into performing a consistent and comprehensive pseudonymization of Hungarian texts, like the system presented here performs.[5] It either left most entities intact (despite an explicit request not to retain any original names or data), or it just used single letters to mask more (still not all) entities in the text.

6 Conclusion

We presented a pseudonymization/data masking prototype for Hungarian providing functions of entity identification and extraction, automatic generation and selection of replacement candidates, and automatic and consistent replacement and reinflection of entities in the final pseudonymized document. The named entity recognition model handles most relevant entity types well, however ID-like entities need to be handled separately to achieve proper performance (not handled in the current prototype). The simple entity embedding model used for replacement candidate generation has some limitations, however, we managed to handle the problem of replacing names consistently to a reasonable degree. Performance of the prototype is acceptable, although further improvement is needed to develop it into a fully-fledged, reliable data masking solution that also outputs completely consistent text with names having a completely natural distribution.

References

1 Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. doi:10.1162/tacl_a_00051.

2 Attila Novák and Borbála Novák. Cross-lingual generation and evaluation of a wide-coverage lexical semantic resource. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL: https://aclanthology.org/L18-1007.

3 Attila Novák and Borbála Novák. NerKor+Cars-OntoNotes++. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2022)*, pages 1907–1916, Marseille, France, June 2022. European Language Resources Association. URL: https://aclanthology.org/2022.lrec-1.203.

4 Attila Novák. A new form of Humor – Mapping constraint-based computational morphologies to a finite-state representation. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1068–1073, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). URL: http://www.lrec-conf.org/proceedings/lrec2014/pdf/207_Paper.pdf.

5 Attila Novák, Borbála Siklósi, and Csaba Oravecz. A new integrated open-source morphological analyzer for Hungarian. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1315–1322, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL: https://aclanthology.org/L16-1209.

[5] We did not perform exhaustive prompt engineering. We tried (the Hungarian equivalent of) the following prompt variants: *a. Replace all names and sensitive data in the text below consistently with a similar name.* b. a+*(not letters)*, c. b+*Do not retain any original names or data.*

6 György Orosz and Attila Novák. PurePos 2.0: a hybrid tool for morphological disambiguation. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 539–545, Hissar, Bulgaria, September 2013. INCOMA Ltd. Shoumen, BULGARIA. URL: `https://aclanthology.org/R13-1071`.

7 Tamás Váradi, Eszter Simon, Bálint Sass, Iván Mittelholcz, Attila Novák, Balázs Indig, Richárd Farkas, and Veronika Vincze. E-magyar – A Digital Language Processing System. In Nicoletta Calzolari (Conference chair), Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12 2018. European Language Resources Association (ELRA).

8 Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. OntoNotes Release 5.0, 2013. `doi:10.35111/xmhb-2b84`.