Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 4802–4810
Marseille, 11–16 May 2020

Much Ado About Nothing
Identification of Zero Copulas in Hungarian Using an NMT Model

Andrea Dömötör[1,2], Zijian Győző Yang[1], Attila Novák[1]
[1]MTA-PPKE Hungarian Language Technology Research Group,
Pázmány Péter Catholic University, Faculty of Information Technology and Bionics
Práter u. 50/a, 1083 Budapest, Hungary
[2]Pázmány Péter Catholic University, Faculty of Humanities and Social Sciences
Egyetem u. 1, 2087 Piliscsaba, Hungary
{surname.firstname}@itk.ppke.hu

Abstract

The research presented in this paper concerns zero copulas in Hungarian, i.e. the phenomenon that nominal predicates lack an explicit verbal copula in the default present tense 3rd person indicative case. We created a tool based on the state-of-the-art transformer architecture implemented in Marian NMT framework that can identify and mark the location of zero copulas, i.e. the position where an overt copula would appear in the non-default cases. Our primary aim was to support quantitative corpus-based linguistic research by creating a tool that can be used to compile a corpus of significant size containing examples of nominal predicates including the location of the zero copulas. We created the training corpus for our system transforming sentences containing overt copulas into ones containing zero copula labels. However, we first needed to disambiguate occurrences of the massively ambiguous verb *van* 'exist/be/have'. We performed this using a rule-base classifier relying on English translations in the English-Hungarian parallel subcorpus of the OpenSubtitles corpus. We created several NMT-based models using different sampling methods and optionally using our baseline model to synthesize additional training data. Our best model obtains almost 90% precision and 80% recall on an in-domain test set.

Keywords: zero copula, syntax, Hungarian, machine learning, transformer model, corpus linguistics, NMT

1. Introduction

Zero copulas, i.e. the phenomenon that nominal predicates lack an explicit verbal copula in some default cases are featured in several languages. In Hungarian, present tense 3rd person indicative is the default case.

(1) a. *János is okos* ***Ø***.
John also clever [zerocop]

'John is clever, too.'

b. *A disznók is okosak* ***Ø***.
the pigs also clever-PL [zerocop]

'Pigs are clever, too.'

c. *János is okos* ***volt***.
John also clever was

'John was clever, too.'

d. *Kételem, hogy János is okos* ***lenne***.
I_doubt that John also clever would_be

'I doubt that John would be clever, too.'

e. *Én is okos* ***vagyok***.
I also clever am

'I am clever, too.'

The purpose of our research was to create a tool that can identify zero-copular constructions and mark the location of the zero copula in sentences like (1a), i.e. the position where the copula would appear in the non-default cases (e.g. 1c).

There are some special constructions in Hungarian where the presence of the verb *van* 'is' is optional. These include some existential/locative constructions (2a-c) with quite strict constraints on the form of specific elements of the construction. E.g. the subject in the existential/locative construction in (2a) must include either the definite or the indefinite article, the locative adverbial must be one of *itt/ott/hol* 'here'/'there'/'where', and the subject must directly follow the place adverbial, otherwise the lack of the verb is not licensed (2b). There are also some special constructions used only in headlines that may lack an overt *van* (2d). We did not consider these as zero copula constructions in our experiments.

(2) a. *Ott (van) a/egy macska!*
there (is) the/a cat

'The cat is there!'/'There is a cat there!'

b. *Ott van János/két macska! (*Ott János/két macska!)*
there is John/two cat

'John is there!'/'There are two cats there!'

c. *Ennek semmi értelme (nincs).*
this-DAT nothing sense (not_have)

'This makes no sense.'

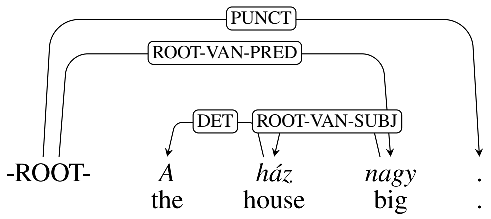
d. *Veszélyben (van) a gázellátás.*
danger-INE (is) the gas_supply

'Gas supply is at risk.'

Our primary aim was to support quantitative corpus-based linguistic research by creating a tool that can be used to compile a corpus of significant size containing examples of nominal predicates including the location of zero copulas. In addition to linguistic research, such a corpus is useful for testing syntactic parsers for Hungarian, since they often fail to parse sentences involving zero copula constructions correctly.

2. Related work

We are not aware of any tool that can process Hungarian text and generate the annotation that we seek. There are dependency parsers for Hungarian, but they do not insert a zero copula into the syntactic annotation they generate. Although the original article (Zsibrita et al., 2013) on the dependency parser later integrated in the `e-magyar` tool chain (Váradi et al., 2018) asserts that dependency trees generated by the parser include a special complex dependency label (see Fig. 3) in the case of nominal predicates, which could be used to identify these constructions, such complex labels are not present in the output of the current version of the `e-magyar` parser.[1] Moreover, even if such labels are present, it cannot be inferred from the output where the copula would be inserted in the non-default tense/mood/person cases.



(3)

Parsers yielding dependency trees following the annotation scheme outlined in version 2 of Universal Dependencies consider the nominal predicate as the head of copula constructions. If there is an overt copula, it is attached to the head by a `cop` arc. Unfortunately, the size of the Hungarian UD corpus is too small (1800 sentences, the training set only 900 sentences) to train a reliable dependency parser.

There is thus no tool that can be used to insert zero copulas, although some tools can be used to identify nominal predicates. As for corpora, the Szeged Dependency Treebank can be mentioned, since it contains zero copulas inserted as virtual nodes into the dependency annotation. However, these nodes were inserted at random positions, sometimes even outside the clause they belong to. Thus sentences containing nominal predicates in this corpus cannot be used as a training set for our purpose either. On the other hand, the Szeged Dependency Treebank can be used as a Gold Standard corpus to evaluate our tool as a classifier: how accurately it can predict the number of nominal predicate clauses in each sentence. The corpus contains 16003 sentences with zero copulas, about 17% of all sentences. We used this empiric ratio when compiling our training and test corpora.

3. Method

We applied a neural machine translation approach to implement our zero copula prediction model. To train a data driven machine translation system one only needs a parallel corpus containing pairs of sentences. In our case, the translation of each sentence is identical to the original, except for those containing nominal predicates. In the latter case a <zerocop> label is inserted at the position of the zero copula.

We used the Marian NMT (Junczys-Dowmunt et al., 2018) framework, an open-source platform implemented in C++.[2] It is easy to install, well documented, memory and resource-efficient. Due to these advantages, it is the NMT tool most often used by researchers and developers.

We used the state-of-the-art transformer (Vaswani et al., 2017) translation architecture and relied on SentencePiece tokenization (Kudo and Richardson, 2018). We used to following parameters and settings when training our models:

- SentencePiece: vocabulary size: 16000; the same vocabulary for source and translation; character coverage: 100%
- Transformer model: 6 encoder and decoder layers; transformer-dropout: 0.1;
- learning rate: 0,0003; lr-warmup: 16000; lr-decay-inv-sqrt: 16000;
- optimizer-params: 0,9 0,98 1e-09; beam-size: 6; normalize: 0,6
- label-smoothing: 0.1; exponential-smoothing

4. Creating the training set

We need a training set that contains explicit labels marking the position of zero copulas. The only corpus available that contains such labels is the Szeged Dependency Treebank. However, the size of that corpus is too small, moreover, the labels are at random positions. Thus we needed to create another training corpus to train our system.

The idea was to transform sentences containing overt copulas (in the past tense) into ones containing zero copula labels by simply replacing the right occurrences of *volt(ak)* by a zero copula placeholder. However, the verb *van* 'exist/be/have' is massively ambiguous. It is used not only as a copula for nominal predicates, but also as a lexical verb for existential (4a), locative (4b), adverbial (4c) and possessive (4d) predication.

(4) a. *Van itt valami.*

There is something here.

b. *Ez itt van.*

This is here.

c. *Ez így van.*

This is so.

d. *Van egy ötletem.*

I have an idea.

[1] Moreover, dependency labels assigned to nominal predicates in coordinated or subordinate clauses are identical to labels assigned to coordinated nominal elements or nominal/adjectival modifiers/adjuncts. Predication is thus in general cannot be distinguished from modification based on the type of dependency relations in the output of the currently available version of the parser.

[2] https://marian-nmt.github.io/

We thus first needed to distinguish instances of overt copulas from instances of lexical *van* 'exist/be (somewhere or somehow)/have'. The verb *van* is considered a lexical verb rather than a copula in these constructions because it must also be overt in the default 3rd person present indicative case. Replacing the past verb form *volt(ak)* by zero would result in an ungrammatical structure in these cases rather than in the present tense equivalent of the sentence.

4.1. A classifier for overt copula sentences

We first tested the performance of the `e-magyar` tool chain at this task on a random 1000-sentence sample containing 3rd person past tense indicative forms (*volt/voltak* 'was/(they) were/existed/had') of the verb *van* of which 598 were instances of nominal predication. The dependency parser of `e-magyar` is supposed to attach the head of the nominal predicate to the verb *van* 'be' by a dependency link labeled `PRED` if the latter is a copula. In other cases, a different type of dependency link is used. Thus the sentences could be simply classified based on the presence of the `PRED` link. Table 1 shows the performance of this dependency-parser-based classifier on the 1000-sentence sample.

Precision	87.8%
Recall	81.0%

Table 1: Performance of the classifier based on the `e-magyar` dependency parser on 1000 sentences

This result indicates that the automatic distinction of copula constructions from other uses of the verb *van* is far from trivial. Some features of Hungarian syntax make this task difficult. One is that functional relations in Hungarian are not expressed configurationally, i.e. grammatical functions cannot be determined based on word order. There are quite strict constraints on the appearance of constituents at dedicated preverbal positions in finite clauses, however whether certain constituents actually occupy these positions or not cannot in general be determined based on the written form of a sentence. This is because of massive ambiguities due to the lack of representation of stress and intonational patterns in writing. There is also a complete lack of (case, number etc.) agreement within NP's: all non-predicative adjectives are in nominative singular whatever the case or number of the NP is unless the head of the NP is a zero pronoun (corresponding to the English pronoun *one*). In the latter case, the final lexical element (e.g. an adjective) is inflected. This combined with the lack of a distinct genitive case (possessors are in general in the nominative case) results in a massive ambiguity of uninflected (non-case-marked) nominal word forms. Yet another feature of Hungarian that makes the identification of nominal predicates difficult is pro drop. This means that unstressed personal pronouns in nominative and accusative case (including pronominal possessors) do not in general have any overt representation. Types of predicates containing a form of *van* cannot thus be distinguished based on the number of uninflected nominal elements in the clause. A nominative noun or adjective in a clause can either be part of a larger NP as an adjunct or possessor, the head of the subject or the head of a nominal predicate. Even if there is just a single noun or adjective in nominative case in a clause, that can be either the subject or the predicate.

As an alternative, we attempted at leveraging more explicit and easier-to-interpret syntactic information present at the English side of translated Hungarian sentences in a parallel corpus to identify sentences containing copulas and nominal predicates. Due to its configurational syntax and the lexical distinction of *be* and *have*, it is easier to identify the types of constructions we want to distinguish based on local features on the English side.

We used a lemmatized, morphologically tagged and disambiguated English–Hungarian parallel corpus (Novák et al., 2019; Novák et al., 2019) with word alignments. The corpus is based on the English-Hungarian subcorpus of the OPUS OpenSubtitles corpus (Lison and Tiedemann, 2016) consisting of 644.5 million English tokens. The English side of the corpus was tagged using Stanford tagger (Toutanova et al., 2003) and lemmatized using morpha (Minnen et al., 2001). A similar processing of the Hungarian side was performed using the PurePos (Orosz and Novák, 2013) tagger and the Humor (Novák, 2014) morphological analyzer. In the analyzed corpus, each original token is represented by at most two tokens: the first token consists of the lemma and the main part-of-speech tag, and this may be followed by a token consisting of additional morphological tags (if any). Tokens of this preprocessed representation of the parallel sentences were aligned using fast align (Dyer et al., 2013). It is advantageous to represent the morphological tags as separate tokens, as it makes it possible for the word aligner to associate certain function words (e.g. English prepositions) that do not have a lexical equivalent with morphological tags on the Hungarian side. The quality of alignment between lexical items is also improved by the reduction of the size of vocabulary due to lemmatization (especially on the Hungarian side).

We selected sentence pairs from this parallel corpus where the Hungarian sentence contains a past tense indicative 3rd person form (*volt* or *voltak*) of the verb *van* (a potential past tense copula). These sentences were classified using a rule-based algorithm.

The algorithm relies on alignments between tokens in the Hungarian sentence and the English equivalent. If *volt(ak)* is aligned with a non-auxiliary *have* or an expletive *there*, the clause is classified as lexical (containing a possessive or existential predicate). If *volt(ak)* is aligned with *be*, it can either be a copula or a lexical verb. In this case, the context is further examined to make a distinction. If *volt(ak)* is not aligned with any of the previously mentioned lexical items, we may assume that the translation did not preserve the clause type and thus the translation cannot be a reliable source of information for classifying the sentence. Sentences like this are marked as skipped.

If *volt(ak)* is aligned with *be*, the algorithm looks for a nominal predicate or a non-nominative argument to distinguish the clause type. These elements are sought in their canonical position (as they do have a canonical position in English in contrast to Hungarian).

First we need to distinguish declarative and interrogative sentences based on the presence of wh-words, a question mark and the position of the verb. In declarative clauses the key element is the first lexical token following *be* that is neither a negation word or a modifier like *more* or *very*. In the case of yes-or-no questions and wh-questions involving *how* and *why*, and (depending on the tags of the tokens following *be*) *what*, *who*, *whose*, and *which* as a question word an extra token is skipped to account for the inversion of word order. Wh-questions involving other wh-words like *where* or *when* are classified as lexical.

(5) a. *Régen ez egy kvalitás volt.*

It used to be **a** quality.

b. *Nem volt otthon.*

He was not **at** home.

(6) a. *Mi volt ez a zaj?*

What was that **noise**?

b. *Miről volt szó?*

What was it **about**?

Decision about the clause type is based on the tag(s) of the token(s) aligned with the key token selected. If it is aligned with and adverb, a postposition or a non-nominative case tag, then the clause is classified as lexical. If it is aligned with a determiner or a nominal lexical item: a noun, an adjective (including participles[3] and comparative or superlative constructions), a numeral or a pronoun in nominative case, then the clause is classified as copular.

The algorithm also applies some lexical rules to handle regular translational equivalences where there is a discrepancy between the type of the English and the corresponding Hungarian construction. One such case is that of constructions describing weather and other environmental conditions (7). While these contain a regular copula on the English side, they contain an expletive *it*, and they correspond to Hungarian constructions that contain an overt *van* in the default present indicative case as well, so they are considered to contain a lexical *van* instead of a copula in Hungarian. The lexical items belonging to this class were obtained performing collocation queries on the Hungarian National Corpus (Oravecz et al., 2014).

(7) a. ***Sötét*** *volt és köd.*

It was **dark** and foggy.

Similar exceptional translational equivalences containing a copula on the English side wile a lexical *van* on the Hungarian side include *igaza van* 'be right' *szerencséje van* 'be lucky', *szükség van* 'be necessary' and *kész van* 'be ready'.

The algorithm classified 458270 sentences containing *volt(ak)* in the Hungarian OpenSubtitles corpus as copular and 332860 as lexical. Its performance was evaluated on the same 1000-sentence test set as that of the classifier based on the `e-magyar` dependency parser. The results are shown in Table 2.

Precision	90,83%
Recall	91,14%

Table 2: Performance of the classifier based on English-Hungarian alignment on 1000 sentences

The results show that our algorithm leveraging syntactic information available in the English translations yielded significantly better precision and recall for classification than the method based on dependency parsing. However, this performance is still below what is to be expected from a tool to be used to generate a gold standard training corpus.

Error analysis revealed that the source of erroneous classification was often not due to the algorithm but to some other factor. Typical sources of error are erroneous tagging and alignment (especially in cases where the sentence contains both copular and lexical constructions involving *volt*) and inexact or erroneous translation. Skipping sentences where instances of *volt* were aligned in an unexpected manner was not completely successful at handling issues stemming from structural or semantic differences between the translation and the original.

4.2. The baseline model

Some classification errors could later be eliminated by applying simple filters to the Hungarian sentence (see below under the description of the *Original improved* model in Section 4.3.) which reclassified about 1% of instances of *volt(ak)* initially erroneously classified as a copula. But in principle we needed to go on with a tool that only works with just above 90% precision to generate our positive training examples for the zero copula recognizer.

We substituted the label `<zerocop>` for overt copulas identified as such by the algorithm outlined above, generating 318843 positive training examples for the machine learning algorithm. As negative training examples, we added 1 million randomly selected sentences form the OpenSubtitles corpus not overlapping with the original sentences from which we generated our positive training examples. We used this training set to train our first baseline model. We also created a development set for training and a test set to evaluate our model (see Section 5. for details).

The first row in Table 3 shows precision, recall and F score figures achieved by our baseline model for clause classification (i.e. the ratio of correctly identified nominal predicates) and those for the exact location matches. The latter are lower because the algorithm may have inserted the right number of zero copulas in the sentence but not at the location in the reference. This model yielded fairly good precision, but poor recall. This is not surprising, since the randomly selected 1 million sentences used as negative training examples (i.e. the ones containing no inserted zero copula placeholders) probably contained many "real" unmarked zero copulas.

[3] There are no complex tenses in Hungarian involving a participle and *van*.

4.3. Improved models

To eliminate as many false negative examples from the training corpus as possible to improve recall, we applied a simple filter to the English side to identify sentences that seem to contain a copula. We were looking for forms of *be* not functioning as an auxiliary verb with a third or second person subject. We needed to exclude sentences with second person subject because polite second person *you* is expressed in Hungarian using pronouns (Ön/Önök/Maga/Maguk) that have third person agreement and undergo pro drop (have no overt realization in nominative or accusative case) like any other personal pronoun. This means that English sentences with a copula and a second person subject are likely to be translated to Hungarian as third person copula sentences with a zero copula. As we aimed at higher recall rather than high precision when trying to identify English copulas, we did not overcomplicate this filter. Although the translation of English copula sentences does not necessarily involve a copula on the Hungarian side and vice versa, this method turned out to be relatively effective in removing the majority of false negative examples from the training set.

We also generated additional positive training examples by running our baseline model on the sentences containing an English copula. We obtained another 161223 positive examples this way.

We created the following models using the training data outlined above:[4]

- **Original filtered**: this model was trained using the 318843 sentences generated from past copula sentences identified by the original classifier algorithm and 1 million randomly selected negative training examples as identified by the simple filter that removed sentences that were identified to contain a copula on the English side. In addition, the negative training data was filtered not to contain one-word sentences and sentences containing rare special characters. These additional filters were meant to remove noisy training data (as the original subtitles corpus is rather noisy).

- **Expanded filtered**: We added the 161223 positive training examples identified by the baseline model (i.e. we had 477082 training sentences that contain zero copulas). To further filter our set of negative training examples, we used our baseline model to mark zero copulas on the set of sentences that were not found to contain a copula on the English side. We used (a subset of) the sentences that were not marked to contain a zero copula by the baseline model as negative training examples in our further models. In this model, we used 2 million negative example sentences.

- **Original improved**: In this model, we tried to eliminate some problems identified in our previous models. E.g. the fact that we did not have some characters in the training data that appeared in our out-of-domain test set (e.g. §) resulted in truncated translations. Thus when creating the training data for this model, we relaxed our character filter. We did not filter one-word sentences from the negative examples either, since this resulted in an overapplication of zero copulas to single-word utterances. We also corrected some easy-to-detect errors in the output of the original overt copula classifier. These error types are illustrated in (8). Overt copulas in Hungarian are always unstressed while instances of lexical *van* 'be/have/exist' are stressed. A clause-initial *volt(ak)* is necessarily stressed (8g). So is one that is preceded by a conjunction (8a,8c) or a relative pronoun (8b), as these are always clause-initial. The word form *volt* also has an adjectival sense 'former/ex' (8d, 8e, 8f). This is also always stressed and usually follows a determiner. Errors of this type were generally introduced by the original classifier algorithm in the case of sentences that contained both a past copula and a stressed *volt* and the latter was also erroneously aligned by the alignment model with the English copula. This model was trained on 314607 positive and 1515204 negative training examples (i.e. this was the first model where the ratio of zero copula sentences in the the training set corresponds to the empirical ratio we found in Szeged Treebank).

- **Expanded improved**: The training set of the previous model was expanded with positive training examples identified by the baseline model and the set of negative examples was also extended to have a ratio of 17% positive examples. This model contained 475830 positive and 2574207 negative examples.

(8) a.

A	*legtöbb*	*az*	*volt,*	*de*	***volt***	*fehér*	*is.*
the	most	that	be-PST-SG3,	but	exist-PST-SG3	white	too

Most of them were like that, but there were some white ones, too.

b.

Az	*a*	*belépőkártya...*	*volt*	*minden,*	*amim*	***volt.***
that	the	access_card	be-PST-SG3	everything	what-POSS.SG1	be-PST-SG3

That access badge... was everything I had.

c.

Házas	*volt,*	*és*	***volt***	*egy*	*fia.*
married	be-PST-SG3,	and	have-PST-SG3	a	son-POSS.SG3

She was married and she had a son.

d.

Megértjük,	*hogy*	*a*	***volt***	*férje*	*és*	*ő*	*üzlettársak*	*voltak.*
understand-PRS-PL1	that	the	former	husband-POSS.SG3	and	she	business_partner-PL	be-PST-PL3

We understand that he and your late husband were business partners.

[4] Training data for all models is available from http://nlpg.itk.ppke.hu/projects/zerokopula.

e. *A* *különleges* *osztag* *egy* ***volt***
the special squad a former
kihallgatótisztje *csinálta.*
interrogator-POSS.SG3 do-PST-SG3_it

It was done by a former Spec Ops interrogator.

f. *Egy* *szinttel* *lejjebb* *voltak* *a*
one level below be-PST-SG3 the
testőrök, ***volt*** *ejtőernyősök* *és*
bodyguard-PL, former paratrooper-PL and
idegenlégiósok.
foreign_legionnaire-PL

On the next floor down were the bodyguards, ex-paratroopers and foreign legionnaires.

g. *Bár* *fiú* *vagy,* ***voltak***
although boy be-PRS-SG2 exist-PST-SG3
jó *válaszok.*
good answer-PL

Although you are a boy, there were (you had) some good answers.

h. *Nem* *olyan,* *mint* ***volt.***
not like_that like be-PST-SG3

It is not like it used to be.

5. Results

We used two test sets to evaluate our model. One was an in-domain test set created from a disjunct part of the same OpenSubtitles corpus we created our training data from. The other test data was derived from Szeged Treebank. The latter contains texts from various domains and genres quite dissimilar from our training data.

Both test sets consist of 2000 sentences, 17% (340) of which contain zero copulas. The test sets were manually checked not to contain erroneous annotation. Positive examples from Szeged Treebank had to be manually edited to move the zero copula markers to their correct position, as they were inserted at random positions in the original corpus. We thus created two golden test sets of identical size and proportions but of different domains and genres.

When testing our models on the test set derived from Szeged Treebank, we faced two problems. One was that of unknown characters (e.g. § in the legal subcorpus). Handling unknown words was a problem to be taken seriously and to be handled in some manner for machine translation systems until focus shifted to the usage of subword lexical units instead of word tokens. Current subword-token-based systems rarely encounter the unknown word/token problem, and there does not seem to be a firm solution anymore to handle it. Our experience was that the transformer model implemented in the Marian toolkit often simply quit generating output at the point it encountered an unknown character (hence an unknown token) in the input, in other cases it substituted some other symbol for it. The `--allow-unk` option of `marian-decoder` did not seem to make any difference. We found long input sentences to have a similar effect. Sentences in the OpenSubtitles corpus are relatively short due to the genre they represent. Szeged Treebank contains much longer sentences. We found that the transformer model seems to be unable to handle input sentences longer than what it was trained on.

We tested all models on both test sets. We performed two types of evaluation on the output. The first one was a more relaxed evaluation of clause classification: i.e. we measured the ratio of correctly identified nominal predicates without checking the position of the inserted zero copula label. The other, more strict, evaluation measured exact zero copula location matches. The results are summarized in Tables 3 and 4. Table 5 shows evaluation results for the full test data (OpenSubtitles and Szeged Treebank combined).

For the in-domain test corpus every model achieved high precision in the classification task. Surprisingly, adding training examples generated by the baseline model to the training corpus (when training the expanded models) resulted in a significant drop in recall. As for identifying the exact copula location, the original filtered model performed best with almost 90% precision and 80% recall. This model yielded the best overall performance on this test data. Our attempts at improving the training data by eliminating some errors in the training data generation process did not result in improvements in performance on the in-domain test set.

On the out-of-domain test set, on the other hand, all models performed far worse. Here, however, adding generated training data and better data filtering seemed to improve performance significantly, especially in terms of precision. Nevertheless, the best expanded improved model achieved only about 70% precision in the classification and 53% in the copula insertion task. This model performed best in almost all respects. Recall was low for all models. This seems mainly to be due to an abundance of sentence types and linguistic patterns missing from the subtitles corpus: very complex sentences with gapped subordination, many coordinated clauses, very long clauses, legal jargon, enumeration of adjectives. Szeged Treebank is dominated by formal written language in contrast to the relaxed oral style present in the OpenSubtitles corpus. The lack of consistent punctuation in the training data may also have had a negative effect on performance.

As for overall performance on the whole test set, we find that improved models obtained better precision but worse recall. Disregarding the baseline model, there is little variance in the overall F_1 scores: there was an overall trade-off between precision and recall.

It is also worth mentioning that the stricter evaluation that considers every zero copula location discrepancy between the reference sentences and the system-generated output an error may be too strict. In Hungarian, word order is relatively free. Word order differences often correspond to differences in topic-focus-comment structure, and have pragmatic function. We manually checked sentences where the models correctly identified the number of zero copulas in the sentence, but the zero copula was inserted at a location different from the reference. As shown in Table 6, in the case of the in-domain corpus the majority of the insertion points was correct for all models. We marked a location implausible when, although the clause is grammatical, it is odd in the given context from a pragmatic point of view.

	Clause classification			Zero copula location		
Model	**P**	**R**	**F1**	**P**	**R**	**F1**
Baseline	**97.83**%	63.56%	77.05%	89.57%	58.19%	70.55%
Original filtered	95.51%	**84.18**%	**89.49**%	**89.74**%	**79.10**%	**84.08**%
Expanded filtered	94.86%	78.25%	85.76%	88.70%	73.16%	80.19%
Original improved	93.61%	82.77%	87.86%	85.62%	75.71%	80.36%
Expanded improved	94.06%	75.99%	84.06%	86.36%	69.77%	77.19%

Table 3: Evaluation results on the OpenSubtitles test set

	Clause classification			Zero copula location		
Model	**P**	**R**	**F1**	**P**	**R**	**F1**
Baseline	52.25%	14.36%	22.52%	35.14%	9.65%	15.14%
Original filtered	52.89%	**29.46%**	37.83%	37.78%	21.04%	27.03%
Expanded filtered	59.58%	28.46%	38.53%	45.08%	**21.53%**	29.15%
Original improved	59.42%	25.74%	35.94%	42.29%	18.31%	25.56%
Expanded improved	**70.48%**	28.96%	**41.05%**	**52.41%**	**21.53%**	**30.53%**

Table 4: Evaluation results on the Szeged Treebank test set

	Clause classification			Zero copula location		
Model	**P**	**R**	**F1**	**P**	**R**	**F1**
Baseline	82.99%	37.34%	51.50%	71.85%	32.32%	44.59%
Original filtered	77.65%	**55.01%**	**64.40%**	67.97%	**48.15%**	**56.37%**
Expanded filtered	80.82%	51.72%	63.07%	71.34%	45.65%	55.67%
Original improved	81.35%	52.37%	63.72%	70.08%	45.12%	54.90%
Expanded improved	**85.40%**	50.92%	63.80%	**73.89%**	44.06%	55.21%

Table 5: Evaluation results on the full test set

We marked agrammatical solutions incorrect (e.g. when the copula landed in a different clause).

6. Error analysis

When reviewing sentences containing erroneously inserted zero copulas, we identified some sentences types that apparently consistently (or at least frequently) resulted in erroneous output. For example, every model tended to insert zero copulas into one-word sentences: **Mióta Ø?* ‘Since when?’, **Elnézést Ø!* ‘Excuse me!’, **Értem Ø.* ‘I see.’, etc. Many one-word sentences do contain a zero copula, and many of them have similar functions/semantics to the sentences into which an overgenerated zero copula was inserted. So this error is to some extent understandable. Especially so for the models the training data of which did not contain one-word negative examples. But this type of error is typical for the other models as well, albeit to a lesser extent.

The models tend to insert zero copulas into discontinuous clauses right at the point where a subordinate clause is inserted into the middle of the clause. The part of the clause preceding the inserted clause does not contain a verb, and the models are urged to do something about it when they see a comma and a conjunction (especially *mint* ‘like/than/as’, (9a)) or a relative pronoun (9b), because this is a typical context where they do occur. *Mint* is a typical element in comparative constructions (*zöldebb, mint* ‘greener than’, *zöld, mint* ‘green like/as green as’). This makes this context a very attractive zero copula landing site.

(9) a. **A boszorkány Ø, aki magához vette a gyereket, eltűnt.*

The witch who had taken the child disappeared.

b. **A hírek valóságtartalma Ø, mint valami méreg, szívódott fel a szervezetébe.*

The veracity of the news was absorbed into his body like some poison.

Another typical error is that all models insert zero copulas into deictic NP’s beginning with *ez a(z)* ‘this’ 10a because this construct is locally ambiguous: *ez Ø a megoldás* ‘this is the solution’ (subject+nominal predicate) vs. *ez a megoldás ... rossz Ø* ‘this solution ... is bad’ (deictic NP), and the *subject+nominal predicate* interpretation is very frequent in the training data.

(10) a. **Mert ez Ø az út a hegy túloldalára vezetett.*

Because this road led to the other side of the mountain.

b. **A mű nem Ø üzletszerű többszörözése és terjesztése a szabad felhasználás körébe tartozik.*

Non-commercial reproduction and distribution of the work is considered free use.

	in-domain test			out-of-domain test		
Model	**correct**	**implausible**	**incorrect**	**correct**	**implausible**	**incorrect**
Baseline	46.67%	13.33%	40.00%	42.86%	14.29%	42.86%
Original filtered	53.85%	7.69%	38.46%	22.22%	22.22%	55.56%
Expanded filtered	66.67%	6.67%	26.67%	36.36%	9.09%	54.55%
Original improved	63.64%	4.55%	31.82%	50.00%	6.25%	43.75%
Expanded improved	70.00%	10.00%	20.00%	38.89%	11.11%	50.00%

Table 6: Is the zero copula location proposed by the model correct in spite of being different from the reference location?

Negation (*nem* 'not' + NP) is a similar structure often misinterpreted by the models, as it contains the same type of structural ambiguity (10b).
Insertion of a zero copula after the first element of a list of adjectives separated by a comma from the next adjective in the sequence is another frequent type of error (11). This error is frequent in the out-of-domain Szeged test set. The construction is not typical of the subtitles corpus (or the comma is often missing in similar constructions), and so the comma is mistakenly interpreted by the models as one separating clauses. Inconsistent use of punctuation in the training corpus seems to be a major source of errors anyway.

(11) **Bivalyszerű Ø, fekete nyakizmai kidagadtak.*

His buffalo-like, black neck muscles bulged.

These structures can be interpreted as predication at the given point of sequential processing, and thus can be considered as psycholinguistically motivated. Other similarly garden-path-like structures include appositive constructions (12a), ellipses and coordinated nominal predicates (12b-12d).

(12) a. **A Budapesti Értéktőzsde részvényindexe Ø, a BUX 40 112,38 ponton zárt kedden.*

Budapest Stock Exchange Index BUX closed at 40,112.38 points on Tuesdays.

b. **És ki tudna ezen változtatni, ha nem te Ø.*

And who could change that if not you.

c. **Az egyik túl nagy Ø, a másik túl kicsi lenne.*

One would be too large, the other too small.

d. **A részvénytársaság igazgatósági tagja Ø, egyben vezérigazgatója lett.*

He became a member of the Board of Directors of the Company and its Chief Executive Officer at the same time.

Some types of errors are typical only of the models obtained by extending the training corpus with positive training examples generated by the baseline model. These models sometimes misinterpreted vocatives as predication (13a), an error not seen in the output of models not trained in synthetic data. Furthermore, these models committed the "most unreasonable" errors. They even inserted zero copulas right after conjugated verbs (13b). This indicates that there is a risk of error propagation and accumulation when we use one model to iteratively generate training data for another.

(13) a. **A kiképzések még nem ért véget, ifjú barátok Ø.*

Your training is not over yet, young friends.

b. **Először sokat ittak Ø, aztán csókolóztak...*

First they drank a lot, then they kissed ...

7. Conclusion

In this paper, we presented a tool for automatically recognizing Hungarian clauses containing nominal predicates and inserting zero copulas into them at the right position (i.e. where an overt copula would be present in the non-default case). On our in-domain test set, our model was able to insert the zero copula into the correct location with nearly 90% precision and 80% recall. We have also tested our models on a test set derived from Szeged Treebank containing a significant amount of legal, literary and news text, which significantly differ from our training data consisting mainly of simple spoken language texts and contain much more complex structures. Consequently, the performance of our models is much lower on this out-of-domain test set, especially in terms of recall. We have also presented an error analysis reviewing typical types of errors of our models.

Acknowledgments

This research 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 as well as through the Artificial Intelligence National Excellence Program (grant no.: 2018-1.2.1-NKP-2018-00008).

8. Bibliographical References

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A Simple, Fast, and Effective Reparameterization of IBM Model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648. Association for Computational Linguistics.

Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Aji, A. F., Bogoychev, N., Martins, A. F. T., and Birch, A. (2018). Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia, July. Association for Computational Linguistics.

Kudo, T. and Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, November. Association for Computational Linguistics.

Lison, P. and Tiedemann, J. (2016). OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Minnen, G., Carroll, J. A., and Pearce, D. (2001). Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223.

Novák, A., Laki, L. J., Novák, B., Dömötör, A., Ligeti-Nagy, N., and Kalivoda, Á. (2019). Creation of a corpus with semantic role labels for hungarian. In *Proceedings of the 13th Linguistic Annotation Workshop, LAW@ACL 2019, Florence, Italy, August 1, 2019*, pages 220–229.

Novák, A. (2014). A New Form of Humor – Mapping Constraint-Based Computational Morphologies to a Finite-State Representation. In Nicoletta Calzolari, et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1068–1073, Reykjavik, Iceland, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L14-1207.

Novák, A., Laki, L. J., and Novák, B. (2019). Mit hozott édesapám? Döntést – Idiomatikus és félig kompozicionális magyar igei szerkezetek azonosítása párhuzamos korpuszból. In *XV. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2019)*, pages 63–71, Szeged. Szeged University.

Oravecz, Cs., Váradi, T., and Sass, B. (2014). The Hungarian Gigaword Corpus. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Orosz, Gy. and Novák, A. (2013). PurePos 2.0: a hybrid tool for morphological disambiguation. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*, pages 539–545, Hissar, Bulgaria. Incoma Ltd. Shoumen, Bulgaria.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich Part-of-speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In I. Guyon, et al., editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Váradi, T., Simon, E., Sass, B., Mittelholcz, I., Novák, A., Indig, B., Farkas, R., and Vincze, V. (2018). E-magyar – A Digital Language Processing System. In Nicoletta Calzolari (Conference chair), et al., 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).

Zsibrita, J., Vincze, V., and Farkas, R. (2013). magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In *Proceedings of RANLP*, pages 763–771.