When Simple n-gram Models Outperform Syntactic Approaches: Discriminating between Dutch and Flemish

Martin Kroon
Leiden University Centre for Linguistics
Leiden University
The Netherlands
m.s.kroon@hum.leidenuniv.nl

Masha Medvedeva
Center for Language and Cognition
University of Groningen
The Netherlands
m.medvedeva@rug.nl

Barbara Plank
Department of Computer Science
IT University of Copenhagen
Denmark
bplank@itu.dk

Abstract

In this paper we present the results of our participation in the *Discriminating between Dutch and Flemish in Subtitles* VarDial 2018 shared task. We try techniques proven to work well for discriminating between language varieties as well as explore the potential of using syntactic features, i.e. hierarchical syntactic subtrees. We experiment with different combinations of features. Discriminating between these two languages turned out to be a very hard task, not only for a machine: human performance is only around 0.51 F_1 score; our best system is still a simple Naive Bayes model with word unigrams and bigrams. The system achieved an F_1 score (macro) of 0.62, which ranked us 4th in the shared task.

1 Introduction

The Dutch language is regulated by the Dutch Language Union. The varieties of Dutch spoken in the Netherlands and spoken in Belgium are both subject to this regulation. Despite this, there are still differences to be found between Netherlandic Dutch and Flemish Dutch, most clearly in phonology and pronunciation, but also in terms of word use and word order. Nevertheless, there is little to no work on automatic classification to distinguish between the two varieties. A first attempt was made by van der Lee and van den Bosch (2017), in light of whose work this year's iteration of the annual Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial) took on *Discriminating between Dutch and Flemish in Subtitles* (DFS) as one of their evaluation campaigns (Zampieri et al., 2018).[1]

For the DFS 2018 shared task, our team (mmb_lct), built a model that discriminates between the two varieties. The model achieved the fourth place out of seven in the ranking of the results – statistical significance of the differences between the F_1 scores of the submissions was taken into account, such that the third place was shared by four teams and we shared our fourth place with STEVENDU2018 (even though their model performed slightly better).

As mentioned, the difference between Dutch and Flemish is most noticeable in spoken language. As we are not dealing with spoken language in this shared task, we are left only with lexical differences and syntactic differences, making the task significantly harder. A problem is that most variety-characterizing words are not all that common. An example would be Dutch *slager* vs. Flemish *beenhouwer* 'butcher' or Dutch *punaise* vs. Flemish *duimnagel* 'thumbtack'. More frequent lexical differences can have another

[1] As the shared task is called *Discriminating between Dutch and Flemish in Subtitles*, we shall refer to Netherlandic Dutch and Flemish Dutch as Dutch and Flemish, respectively. It should be noted that, in this task, Dutch and Flemish are both the standard varieties, as regulated by the Dutch Language Union, and not dialects.

Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects, pages 244–253
Santa Fe, New Mexico, USA, August 20, 2018.

problem, where the words occur in both varieties, but are used differently: Dutch *mooi* vs. Flemish *schoon* 'beautiful' are more frequent, but *schoon* also occurs in Dutch, meaning 'clean'.

Syntactic differences are found mostly in verbal clusters, where the two varieties show different word orders – or at least, a preference for certain word orders. For example, in Flemish the word order in sentence-final verbal clusters is modal–main verb–perfective auxiliary (*moet gemaakt hebben* lit. 'must made have'), while in Dutch the orders modal–perfective auxiliary–main verb and main verb–modal–perfective auxiliary are both widespread (*moet hebben gemaakt* 'must have made' and *gemaakt moet hebben* lit. 'made must have', respectively). Flemish also, for example, allows for the interruption of these types of verbal clusters by adpositions, adverbs or even nominal objects, whereas Dutch is much less likely to exhibit this syntactic behaviour (Barbiers et al., 2005; Barbiers et al., 2008).

In order to quantify these syntactic differences, we used syntactic subtrees of dependency parses to distinguish between the two varieties, which is, to the best of our knowledge, a novel approach in language identification (Jauhiainen et al., 2018). As opposed to n-grams, dependency subtrees allow us to detect non-contiguous patterns of words as well as to determine syntactic relations between the words. This is useful as this also, for example, allows us to quantify whether one of the two varieties is more inclined to topicalize the object of a verb.

We decided to approach the DFS 2018 shared task by mainly focusing on the use of linear classification algorithms, as they steadily seem to outperform neural approaches in the task of language identification (Zampieri et al., 2017; Malmasi et al., 2016; Zampieri et al., 2015; Zampieri et al., 2014). In the next Section we discuss relevant previous research in the discrimination between similar languages. In Section 3 we describe the data released for the DFS shared task, the features we used as well as our system submissions. Section 4 follows with the results, which are discussed in Section 5.

2 Related Work

Although this year is already VarDial's fifth anniversary (Zampieri et al., 2018; Zampieri et al., 2017; Malmasi et al., 2016; Zampieri et al., 2015; Zampieri et al., 2014), it is the first time that the DFS shared task was organized. Previous iterations of VarDial saw shared tasks concerning Arabic dialect identification, German dialect identification, cross-lingual dependency parsing, and discrimination between similar languages (DSL).

Last year our team participated in the DSL shared task (Medvedeva et al., 2017), where it ranked second with an F_1 score of 0.925. Bestgen (2017) won by 0.002 points. Both systems used a two-layer classification, comparable to Goutte et al. (2014): the first layer identified the language group, the second layer trained multiple classifiers to identify language varieties within the groups.

However, while we only used word and character n-grams in both classification layers, the winning team, Bestgen (2017), also used POS-tags n-grams (the POS tags were obtained using language-group specific POS taggers) and some global statistics, such as proportion of capital letters and punctuation marks, as features in the second layer. This is in line with van der Lee and van den Bosch (2017), who, next to word n-grams, use global statistics and POS-tag n-grams in the classification of Dutch vs. Flemish subtitles, achieving an F_1 score of 0.92. For this task we depart from our earlier system but explore alternatives, in particular one based on using syntactic information.

Even though POS-tagging is a challenging method to use for language identification due to its language-dependent nature, it has often been explored for distinguishing between language varieties (Martinc et al., 2017; Adouane and Dobnik, 2017, among others). To add to previous experiments that exploited POS-tag n-grams and thus retaining the linear structure of the text, we explore the possibilities of using hierarchical subtrees to allow for non-contiguous groups of POS tags as well as to exploit the syntactic relations between them.

3 Methodology and Data

In this section we describe the DFS data as it was released to participants, the features we used in our approaches, and our three system submissions.

FLEMISH	DUTCH
Sami, DiMera, Amai, Sooz, interesseerd, Kenshee, Moz, vanop, AUDIO, Megatron, Shishio, enant, ACHTERGRONDMUZIEK, Celeste, ente, komaan, Jumanji, Kiriakis, yardlijn, Breanna	MUZIEK, EEN, Oke, STEM, Text, LACHT, ZE, GEJUICH, LACHEN, AFV, SPANNENDE, Broadcast, Funniest, Brainiac, Piha, surveillanten, BEAU, oke, Biaggi, MUZIEKJE

Table 1: Most frequent words in each language variety.

3.1 Data

The data consisted of professionally produced subtitles for Dutch and Flemish television. The gold label – i.e. whether the fragment was Dutch or Flemish – was based on the country for which the subtitles were created. If the subtitles were shown on a Dutch TV network, it was labelled Dutch; if they were shown on a Flemish TV network, it was labelled Flemish. As for distribution of genres between Dutch and Flemish in this dataset, it is fairly similar (van der Lee and van den Bosch, 2017).

In particular, the training set consisted of 300,000 fragments with an average length of 40.62 tokens, including punctuation. The set was evenly balanced, such that 150,000 fragments were Dutch and 150,000 fragments were Flemish (with an average of 40.69 and 40.56 tokens per fragment, respectively). A fragment may contain multiple sentences, averaging at 5.52 sentences per fragment. The vocabulary for Dutch consisted of 127,546 tokens, for Flemish it consisted of 120,050 tokens. Out of those, 62,758 Dutch tokens only occurred in Dutch, while 55,262 tokens in the Flemish vocabulary only occurred in Flemish fragments. The 20 most frequent words in the training data that only occurred in one language variety are shown in Table 1. The development set was significantly smaller, consisting of only 500 fragments (on average 40.58 tokens and 5.64 sentences per fragment).

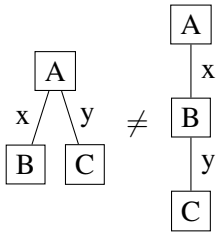
The test set, which was withheld, consisted of an evenly balanced set of 20,000 fragments. In terms of average number of tokens and sentences, the test set is very similar to the training an development sets: 40.60 and 5.54, respectively.

It was noted, however, that the training set and the development set contained several encoding errors. For example, the third fragment of the training set contained *financi le*, where an *ë* is missing (intended was *financiële* 'financial'), leading to two non-existing words *financi* and *le*. Although this might not be of much influence when using character n-grams, it will be of influence when using word n-grams, when POS-tagging or when parsing syntactically. Our team made no attempt, though, to mitigate these encoding errors by, for example, using a spell checker to fill in the missing characters.

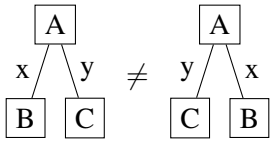
The first author of this work, as a native speaker of (Netherlandic) Dutch, also noted how hard it was in this particular data set to manually classify the subtitles. Very few specifically Dutch and Flemish words were used (although there were plenty of words that were only used in Dutch or only in Flemish, most of these words are not necessarily characterizing for Dutch or Flemish – this is illustrated in Table 1, where only *komaan* 'come on' is typically Flemish; most Dutch-only words are commentary for the hearing-impaired (written in capitals), such as *MUZIEK* 'music'), nor were there many specifically Dutch or Flemish syntactic constructions that the first author recognized. As a test on 25 randomly selected fragments from the training data, the first author performed on chance level with an F_1 score of only 0.51. This difficult character of the task was also reflected in the results of the DFS shared task: the system of the winning team achieved an F_1 score of 0.66.

3.2 Features

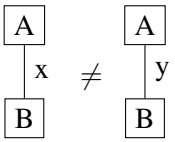
As features we mostly resorted to word and character n-grams, as motivated by our submission to last year's DSL shared task. We also present a novel approach to language classification which relies on subtrees of dependency parses as features. This was motivated by the fact that we can use one Dutch parser model, since standard Dutch and standard Flemish are sufficiently similar. Of course, when one needs to classify between two languages (as opposed to two – very similar – varieties), one cannot easily



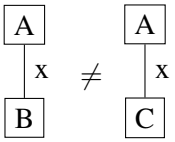
(a) Nodes need to be connected in the same way.



(b) The order of sister nodes must be the same.



(c) Labels of corresponding edges need to be the same.



(d) Corresponding nodes need to have the same label.

Figure 1: Isomorphism constraints of subtrees.

use parsers, since the parser models are language-specific.

In order to use subtree features, we parsed the data using a Universal Dependencies parser implemented in UDPipe (Straka and Straková, 2017), specifically using the CoNLL17 Shared Task Baseline UD 2.0 Model for Dutch.[2] Dependency parses of fragments were represented as a directed acyclic graph in the Python package `networkx` (Hagberg et al., 2008), with every sentence in the fragment being its own connected component in the graph. Graphs were also ordered, meaning that the linear order of sister nodes was retained. The order of words relative to their head was not necessarily retained: verbs with a direct object to their left were not automatically distinct from verbs with a direct object to their right – we experimented with the option of retaining this relative order of words to their heads. From these graphs, subtrees were extracted, counting for every fragment which subtrees occurred in them and how often.

A subtree was defined as any combination of n nodes in the dependency tree that form a connected component. Nodes (that is, words) were represented as POS tags. Therefore, these subtrees can be considered hierarchical POS-tag n-grams containing syntactic relations; whereas normal n-grams are contiguous sequences, these subtrees are not necessarily.

Two subtrees were considered to be isomorphic if they contained the same amount of words, the words were connected in the same way with the same syntactic relation, the corresponding words between the two subtrees had the same POS tags, and the order of sister nodes was the same. This is illustrated in Figure 1.

Bare POS tags in Universal Dependencies (which are 17 coarse-grained categories) can be quite unspecific: there is no distinction between finite verbs and past participles. Therefore we experimented with using morphological features (as tagged by UDPipe) as well, effectively adding another isomorphism constraint that requires all morphological features of corresponding nodes to be the same as well (making that infinitives are not considered the same as third person verbs, for example).

3.3 Systems

We submitted three runs with different classifiers and features. We only focused on linear classifiers, specifically, Support Vector Machines with a linear kernel (LinearSVC) and a Naive Bayes classifier.[3] We chose which systems to submit based on cross-validation and evaluation on the development set.

3.4 Run 1

For the first run we used a linear SVM with word and character n-grams, which has been proven to work for language identification between similar languages before (Medvedeva et al., 2017; Rama and

[2] `http://ufal.mff.cuni.cz/udpipe/users-manual`

[3] As implemented in `scikit-learn` (Pedregosa et al., 2011).

Çöltekin, 2017). We use the same features as in our earlier participation in a related VarDial shared task: word uni- and bigrams and character n-grams with 1 to 6 characters. As opposed to last year, we have though not used tf-idf weighting, as it has shown to yield lower results. Regarding preprocessing, we split the data into tokens with a simple multilingual tokenizer[4] that uses whitespaces as the main reference point. Punctuation was not separated from the words, nor have we lowercased the text.

We found it performed with mean 62.1% accuracy using 3-fold cross-validation and 69% accuracy on the development set. Despite being very similar to the system used in Medvedeva et al. (2017), which is seemingly language-independent and performed very well in the task of distinguishing between Bosnian, Croatian and Serbian, these results indicate that distinguishing between Flemish and Dutch is a much harder task. A confusion matrix can be found in Table 2.

	predicted: BEL	predicted: DUT
true: BEL	93291	56709
true: DUT	56948	93052

Table 2: Confusion matrix for 3-fold cross-validation for Run 1 - Linear SVM.

Additionally, we have plotted the top coefficients for both classes in Figure 2. The coefficients show once again that the data hardly contain any characterizing words for the two varieties: only Flemish interjections such as *Allee,* 'come on', *Komaan,* 'come on' and *Wel,* 'well' are very characterizing, as is the bigram *naar hier* 'to here'. Other predictors are not as typical: for example, although Dutch *MasterChef* has 114 instances in the training data for Dutch and 2 for Flemish, that only suggests that a MasterChef TV-show was included in the Dutch part of the dataset.

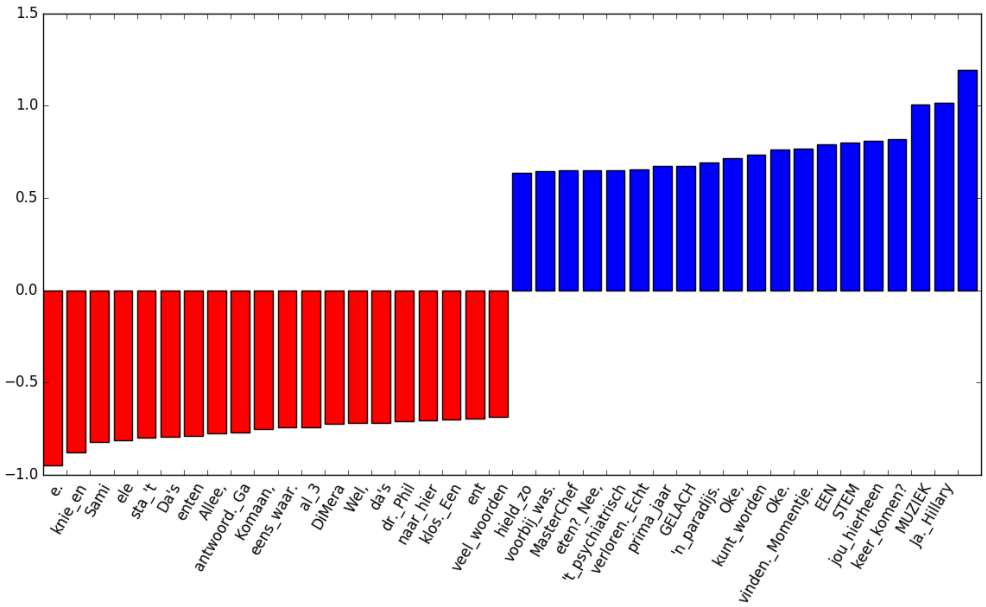


Figure 2: Coefficients (weights) as assigned by the first run's SVM to the two varieties. The top 20 predictors for Flemish are on the left (red) and the top 20 predictors for Dutch are on the right (blue).

[4] https://github.com/bplank/multilingualtokenizer

3.5 Run 2

For the second run we also used a linear SVM, with hierarchical POS-tag subtrees, as decribed in Section 3.2. We used ordered subtrees of sizes 1 and 2, retaining the relative position of words to their head, but ignoring morphology. Including larger subtrees showed lower results (i.e. 55% vs. 57% accuracy on the development set). We also experimented with using tf-idf weighting, but this also resulted in a worse performance. Using cross-validation, the system achieved 55% accuracy. A confusion matrix for the run can be found in Table 3.

	predicted: BEL	predicted: DUT
true: BEL	95161	54839
true: DUT	80045	69955

Table 3: Confusion matrix for 3-fold cross-validation for Run 2 - Syntactic Subtrees.

Since these features didn't include any information on the words themselves, but only on the POS tags, our hope was that if we combine the syntactic information with the features from the first run, we will be able to pick up on much more differences between the dialects. However, the results showed the opposite. With a combination run we achieved an accuracy of only 54%, which means that adding syntactic information only hurts performance.

3.6 Run 3

Our third run was a simple Naive Bayes system that used word uni- and bigrams. In this model we lowered case and used a built-in `scikit-learn` tokenizer (Pedregosa et al., 2011).

The model achieved 63% accuracy when evaluated using 3-fold cross-validation. From the confusion matrix for cross-validation results in Table 4 we can see that Dutch is confused more often than Flemish. Moreover, on the development set this classifier performed with a 68% accuracy. This made it a good contender for our first run, which got 69%.

	predicted: BEL	predicted: DUT
true: BEL	101993	48007
true: DUT	62672	87328

Table 4: Confusion matrix for 3-fold cross-validation for Run 3 - Naive Bayes.

4 Results

The results on the official evaluation data are shown in Table 5 and confirm our findings on the development data. Our best system is the simple Naive Bayes classifier, reaching an F_1 score of 0.62. It is closely followed by the our first run (0.61). Our syntactic subtrees reached a much lower F_1 score of 0.49. Our results corroborate earlier findings where a Naive Bayes outperformed alternative approaches (Tiedemann and Ljubešić, 2012). We also investigated their blacklist approach, but in preliminary experiments on the development data it resulted in below-chance performance. The Naive Bayes approach is slightly more robust on this task, as shown by comparing a cross-validation to a single official dev split setup, on which the Naive Bayes dropped less than the more overfitting-prone higher capacity SVM approach.

Overall our approach ranked 4th in the shared task, with most systems ranking around 0.63, and the winning system reaching a top performance of 0.66.

We also saw that Dutch is more often wrongly classified as Flemish than Flemish as Dutch by our best system, suggesting that, based on our features, it is harder to correctly classify Dutch than Flemish.

System	F1 (macro)
Random Baseline	0.5000
NAIVE BAYES	**0.6201**
SVM	0.6105
SYNTACTIC SUBTREES	0.4895

Table 5: Results of our submissions on the official DFS task test data.

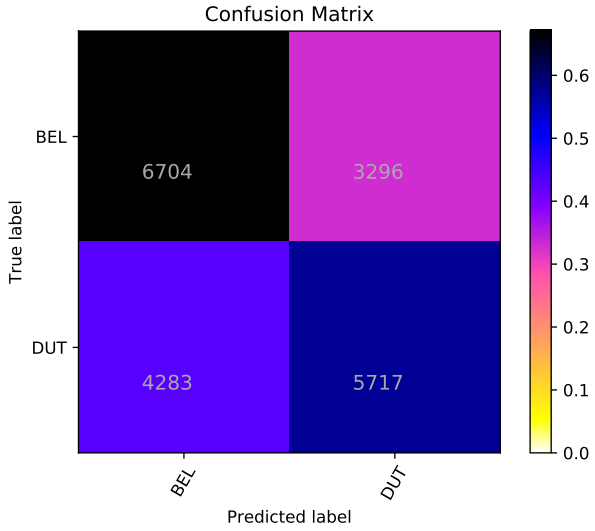


Figure 3: Confusion matrix of our best system (run 3) on the test data provided by the organisers.

This is shown in the confusion matrix in Figure 3. This is contrary to what van der Lee and van den Bosch (2017) found, who found that Flemish was harder to classify. They had though a different setup, in particular an imbalanced data set, which surely had an influence on the results.

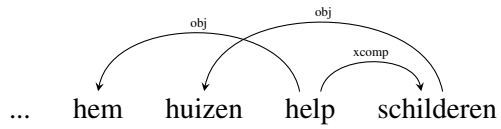
5 Discussion

The use of ordered syntactic subtrees (hierarchical POS-tag n-grams containing syntactic relations, if you will) in the automatic identification between Dutch and Flemish, then, does not seem to help – in fact, it influences results negatively: a classifier that uses word and character n-grams alone performs significantly better than one that uses a feature union between n-grams and syntactic subtrees. There are several possible explanations for this.

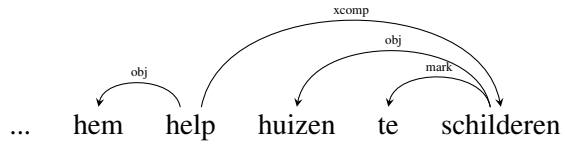
A first explanation can be the performance of UDPipe in general. The labelled attachment score (LAS)[5] of Dutch parses is about 70% to 80% (Straka and Straková, 2017). Errors in the parses may have led to noisy features. Additionally, the already addressed encoding errors in the data as well as the frequent commentary for the hearing-impaired will have led to more incorrect parses, leading to more noisy features.

Secondly, it may be the case that (standard) Dutch and (standard) Flemish are simply not sufficiently distinct syntactically in terms of simple POS tags. As described above, we did try using morphological features, such that finite verbs can be distinguished from infinitives or participles. However, using morphological features resulted in lower performance. It may be that using morphological features made the POS tags too specific, as it also distinguishes singular nouns from plural nouns, for example. It was not tested in this work if perhaps using certain combinations of morphological features (as opposed to using all or none) do yield better results. This is certainly worth looking into in future research.

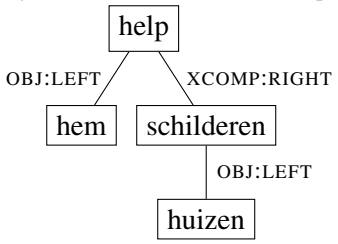
[5]The LAS measures the percentage of words that are assigned both the correct syntactic head and the correct dependency label, i.e. syntactic relation. Unlabelled attachment scores, where the correct head is assigned though with a wrong label, are usually somewhat higher (about 5%), but because our isomorphism constraints require all edge labels to be identical, we need the LAS.



(a) A sentence fragment with a cross-serial dependency relation.



(b) A sentence fragment without a cross-serial dependency relation.



(c) The `networkx` representation of Figures 4a and 4b, ignoring the presence of *te* 'to'.

Figure 4: An illustration of how the presence of cross-serial dependency relations in a fragment is lost when converting to `networkx` subtrees. The relative order of sister nodes is retained, as is the relative order of words to their head using a tag on the syntactic relations. Because the relative order of nodes to the head of their head is not retained, we can no longer distinguish between 4a and 4b. The fragment means '... help him paint the house'.

Lastly, the size of the subtrees that we extracted could also be of influence. Having looked only at subtrees with one, two or three words in them, we ignored larger subtrees that may in fact be more informative in the task of automatic classification of Dutch vs. Flemish. We deem this unlikely, though, given that the frequencies of subtrees plummet as the size increases – comparable to n-grams. At the same time the amount of possible subtrees of size n that can be extracted from a fragment increases faster than the amount of possible n-grams (of size n) due to a multitude of distinct dependency labels, in addition to the fact that the words in a subtree need not be string-adjacent but only need to form a connected component. This decreases the average frequency of subtrees even further as the amount of words in them grows.

On a different note concerning parses, the way syntactic subtrees are represented in `networkx` in this work, does not support for cross-serial dependency relations, which are prevalent in Dutch (Bresnan et al., 1982): although it does retain the relative order of sister nodes and optionally of nodes to their mothers, it never retains the relative order of nodes to their grandmothers (i.e. the head of its head). This results in the impossibility to distinguish between the construction in Figure 4a, which shows a cross-serial dependency relation, and the construction in Figure 4b, which does not. In this work Figure 4a and Figure 4b yield identical subtrees, ignoring the presence of *te* 'to' in Figure 4b; this is illustrated in Figure 4c. If the relative order of *huizen* to *help*, which is its grandmother node, were retained, the subtrees would have been distinct, as *help* is on the left of *huizen* in 4a, whereas in 4b it is on its right. When there is, then, a strong difference in the usage of these two constructions between Dutch and Flemish, this information is lost. It will be interesting for future research to see if adequately representing cross-serial dependency relations in subtrees will influence the performance of a Dutch-Flemish classifier.

The use of feature selection in our subtree approach should certainly be explored in future work. As mentioned before, the amount of noisy subtree features was probably quite high. By reducing these noisy features, the weight assignment to more informative features can thus be boosted, yielding better predictors. Setting a simple minimum-frequency constraint for features, for example, could improve results, in line with Bestgen (2017), who only uses character n-grams that occur at least 100 times.

We also did not explore the use of subtrees with algorithms other than linear SVMs. It would be interesting to see if different results can be achieved with a non-linear SVM, a neural approach or a decision tree. It would also be interesting to experiment with a blacklist approach (Tiedemann and Ljubešić, 2012) applied to subtrees.

Although the dependency-subtree approach as proposed in this work is outperformed by traditional n-gram models, there are still many options to try and improve its performance, such as morphological fine-tuning, using words instead of POS tags, changing the isomorphism constraints, implementing support for cross-serial dependency relations, feature selection and different classification algorithms. We leave these suggestions to future research.

As a final note, we had a few concerns about the data: van der Lee and van den Bosch (2017) mention that the gold labels were based on the country where the program was broadcast, but whether subtitles are broadcast in the Netherlands or in Belgium does not necessarily imply that they were produced by a Dutchman or a Fleming. Moreover, professional subtitlers often try to avoid specifically Dutch or specifically Flemish language (which is also supported by the words that occur only in one variety; see Table 1), making the task particularly hard for subtitles. Nevertheless, it was shown by van der Lee and van den Bosch (2017) as well that the task can be done with a high performance, despite the subtitular nature of their data. In fact, this DFS shared task uses another distribution of their data, however it is unclear what exactly causes such a vast difference between their performance and the performances in this shared task.

6 Conclusion

We presented our participation in the VarDial 2018 shared task on discriminating between Dutch and Flemish in subtitles. We investigated both traditional n-gram based models and a syntactic approach. Our results show that the simplest model with the simplest feature set (Naive Bayes with word n-grams) outperforms more involved approaches, in particular our dependency-tree approach, which only performed around chance level. The task turns out to be rather difficult, as shown by the relatively low results among all participating teams in the DFS shared task and the difficulty in a preliminary manual investigation.

References

Wafia Adouane and Simon Dobnik. 2017. Identification of languages in Algerian Arabic multilingual documents. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 1–8.

Sjef Barbiers, Hans Bennis, Gunther De Vogelaer, Magda Devos, and Margreet van der Ham. 2005. *Syntactische Atlas van de Nederlandse Dialecten Deel I / Syntactic Atlas of the Dutch Dialects Volume I*. Amsterdam University Press.

Sjef Barbiers, Johan van der Auwera, Hans Bennis, Eefje Boef, Gunther De Vogelaer, and Margreet van der Ham. 2008. *Syntactische Atlas van de Nederlandse Dialecten Deel II / Syntactic Atlas of the Dutch Dialects Volume II*. Amsterdam University Press.

Yves Bestgen. 2017. Improving the character ngram model for the DSL task with BM25 weighting and less frequently used feature sets. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 115–123, Valencia, Spain, April.

Joan Bresnan, Ronald M. Kaplan, Stanley Peters, and Annie Zaenen. 1982. Cross-serial dependencies in Dutch. In *The formal complexity of natural language*, pages 286–319. Springer.

Cyril Goutte, Serge Léger, and Marine Carpuat. 2014. The NRC system for discriminating similar languages. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects (VarDial)*, pages 139–145, Dublin, Ireland.

Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. 2008. Exploring network structure, dynamics, and function using NetworkX. In Gäel Varoquaux, Travis Vaught, and Jarrod Millman, editors, *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pages 11–15, Pasadena, CA USA, August.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2018. Automatic language identification in texts: A survey. *arXiv preprint arXiv:1804.08186*.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. 2016. Discriminating between similar languages and Arabic dialect identification: A report on the third DSL shared task. In *Proceedings of the 3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan.

Matej Martinc, Iza Škrjanec, Katja Zupan, and Senja Pollak. 2017. PAN 2017: Author profiling – gender and language variety prediction. In *Working Notes of CLEF*, CEUR Workshop Proceedings. CEUR-WS.org.

Maria Medvedeva, Martin Kroon, and Barbara Plank. 2017. When sparse traditional models outperform dense neural networks: the curious case of discriminating between similar languages. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 156–163, Valencia, Spain, April.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct):2825–2830.

Taraka Rama and Çağrı Çöltekin. 2017. Fewer features perform well at native language identification task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 255–260, Copenhagen, Denmark.

Milan Straka and Jana Straková. 2017. Tokenizing, POS tagging, lemmatizing and parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, August. Association for Computational Linguistics.

Jörg Tiedemann and Nikola Ljubešić. 2012. Efficient discrimination between closely related languages. *Proceedings of COLING 2012*, pages 2619–2634.

Chris van der Lee and Antal van den Bosch. 2017. Exploring Lexical and Syntactic Features for Language Variety Identification. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 190–199, Valencia, Spain.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, and Jörg Tiedemann. 2014. A report on the DSL shared task 2014. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects (VarDial)*, pages 58–67, Dublin, Ireland.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Preslav Nakov. 2015. Overview of the DSL shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, pages 1–9, Hissar, Bulgaria.

Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. Findings of the VarDial Evaluation Campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Valencia, Spain.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, Chris van der Lee, Stefan Grondelaers, Nelleke Oostdijk, Antal van den Bosch, Ritesh Kumar, Bornini Lahiri, and Mayank Jain. 2018. Language Identification and Morphosyntactic Tagging: The Second VarDial Evaluation Campaign. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, Santa Fe, USA.