Universiteit
Leiden
The Netherlands

Towards the automatic detection of syntactic differences

Kroon, M.S.

Citation
Kroon, M. S. (2022, November 10). *Towards the automatic detection of syntactic differences*. *LOT dissertation series*. LOT, Amsterdam. Retrieved from https://hdl.handle.net/1887/3485800

Version:	Publisher's Version
License:	Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden
Downloaded from:	https://hdl.handle.net/1887/3485800

Note: To cite this publication please use the final published version (if applicable).

CHAPTER 5

Discussion and conclusion

Over the course of this dissertation I have researched the question of whether it is possible to automatically detect syntactic differences and, if so, how. Before concluding and answering that question, I will briefly summarize the findings of each Chapter, and discuss the findings of all the Chapters in their respective relative contexts.

5.1 Brief summary of previous Chapters

In Chapter 2 the issue of syntactically incomparable sentence pairs was addressed. In parallel corpora it is not a given that sentences that are aligned to one another are syntactically comparable, as they may exhibit vastly different constructions or a free translation. A method and measure was needed to filter out sentence pairs that are syntactically too different, because using free translations, wrongly aligned sentence pairs or translations that are structurally too different for the detection of syntactic differences between the two languages can influence the results negatively.

To this end, four different filtering approaches (one based on the sentence-length ratio, one based on the Levenshtein distance on POS tags, one on the graph edit distance (GED) on dependency parses and one that combines the previous three filters in a regression model) were explored. The results of the ex-

periments on datasets of English, Dutch and German parallel sentences suggest chiefly that filtering for syntactic comparability is a hard task, in part because syntactic comparability is hard to define, which interacts with the trade-off between cleaner data and losing desired variation. The fact that the task is hard was also corroborated by the only moderate inter-annotator agreement, which ranged between 0.61 and 0.26. Nevertheless, the presented filters are useful tools for automatizing the selection of syntactically comparable sentences from a parallel corpus. The filtering approach that combines the other three filters works best, however it requires the existence of a pre-labelled dataset on which it can be trained, is computationally expensive and has a high risk of overfitting on the dataset. In general it was observed that, as expected, using syntactic information (of any kind) gives better results: the Levenshtein distance and the GED outperform the sentence-length ratio. The robustness in its parameters throughout the language pairs furthermore suggested that the GED approach can be used as a default filter, especially when a pre-labelled dataset is not available. This would make sense, as the GED filter uses the most syntactic information and is less sensitive to phrases or constituents transposing. The Levenshtein distance can also give reasonable results, but is expected only to perform well on closely related language pairs, in which the word order is more or less similar.

In Chapter 3 I presented a systematic approach to detect and rank hypotheses about possible syntactic differences for further investigation by leveraging parallel data and using the Minimum Description Length (MDL) principle, which provides an elegant paradigm to find structure in data (among others Grünwald 2007; Barron, Rissanen and Yu 1998). The approach deploys the MDL-based pattern mining algorithm SQS ('Summarising event seQuenceS'; Tatti and Vreeken 2012) to extract sequences of POS tags that can be considered 'typical' syntactic building blocks of a language. From the lists of these POS patterns of two languages, a shortlist of potential syntactic differences is created based on the number of parallel sentences with a mismatch in pattern occurrence. The patterns are then ranked on a χ^2 value calculated from these mismatch frequencies, generating hypotheses on where syntactic differences may be found within the language pair.

The approach was evaluated on parallel corpora of English, Dutch and Czech, and proved useful in both retrieving POS building blocks of a language, which can already be of use to detect broad typological characteristics, as well as pointing to meaningful syntactic differences between languages. Apart from that, with the approach it is possible to detect tagging inconsistencies between two languages easily. It was however observed that the approach is very sensitive to tagging quality, with tagging inconsistencies between languages (i.e. different conventions or annotation guidelines) and tagging inaccuracies within languages (i.e. tagging errors) heavily influencing results. Despite this clear sensitivity to tagging quality, our results and approach are promising, with many hypotheses being generated by the algorithm that proved to be correct.

In Chapter 4 a different approach was explored to detect morpho-syntactic

differences that is, unlike the MDL approach of Chapter 3, not dependent on the availability of natural language processing (NLP) tools for both languages under investigation. The key question of Chapter 4 was whether it is possible to use fully annotated text in language A (called the source language) to detect grammatical properties of a different, less well-described language B (called the target language), and differences between the two languages, in parallel text. To this end, word alignment is used to map source language words to target language words with the aim of detecting syntactic features of the target language and differences between source and target language by semi-automatically analysing this mapping. Three tools were developed to detect syntactic properties and differences from parallel data aligned on a word level: the Data Grouper for Attribute Exploration (DGAE), the Generalization Tree Inducer (GTI), and the Affix-Attribute Associator (AAA). These three tools were evaluated on the language pair English-Hungarian. With the help of the tools 43 hypotheses on morpho-syntactic features of Hungarian or differences between it and English were generated. The hypotheses were independently checked by a native speaker and expert of Hungarian and its syntax, and cross-checked with a list of characteristic differences between Hungarian and English independently compiled by said expert. It was concluded that the tools can be used very effectively to form many correct hypotheses on differences between the languages in several syntactic domains. With the help of the tools, I even generated two hypotheses of which the correctness is yet to be investigated, highlighting the power of the tools in the search for syntactic differences between languages.

5.2 Relating the filter to MDL and alignment

In Chapter 3 we have experimented with the influence of the filter from Chapter 2 on the results from the automatic detection of syntactic differences. It was observed that the Europarl corpus (Koehn 2005), of which a fragment was used in Chapter 3, suffered from free translations and wrongly aligned sentence pairs, which led us to believe that the filter could be deployed successfully.

The design of the filters made it such that the combination, i.e. regression, filter requires a training set of sentence pairs binarily labelled for syntactic comparability, and that the other three filters use a threshold value, which can be set manually or with the use of a grid search on a training set. However, because there was no pre-labelled data set on which the filter could be trained for the purposes of the research of Chapter 3, there was no possibility to deploy the combination filter, that had been found to work best, or to do a grid search for the other three filters. Instead, the GED-based filter was used with a threshold value of 4, which was already suggested as a possible default value for the GED filter in Chapter 2.

The results of the experiments with the filter in Chapter 3 show that using the filter does indeed influence the results. First and foremost, applying the filter results in a significant loss of data. After filtering out incomparable sen-

tence pairs using the GED-based filter, only about one fifth or one sixth of the sentences remained in the data.

This strong reduction of data is probably due to a three-way interaction. The first factor is simply the noisiness of the data: there are a significant number of sentences that are wrongly aligned in Europarl, and an even larger number whose translations are too free for the purposes of comparative-syntactic research. These sentence pairs we wanted to filter out. The second factor is that the filter, not unlike the MDL approach itself, is sensitive to tagging errors and inconsistencies: if a label is incorrect, the edit distance between the two sentences will be higher, which may push the sentence pair over the threshold and have it be discarded wrongly. The last factor is that it may be the case that the threshold value of 4 is not appropriate for the dataset used. Since a training set was not available for the setting of the threshold, however, we had to resort to parameters that were shown to work well in Chapter 2 for a filter that was hypothesized to be robust throughout different language pairs.

It was furthermore observed that the filter had only a marginal effect on the quality of the output of the MDL approach. Filtering resulted in somewhat more useful hypotheses on syntactic differences between English and Dutch, as it reduced that number of patterns ranking highly due to tagging issues. As for the Czech runs, the opposite was true. While for the comparison between Dutch and Czech the difference seemed insignificant, for the comparison between English and Czech the number of useful patterns went down and it strikingly made the approach unable to detect that Czech does not have articles. Nevertheless, filtering the data makes the patterns easier to interpret, because they are generally shorter and contain less noise.

The filter was not deployed in Chapter 4. This is because the combination filter, GED-based filter and the Levenshtein-distance filter require the availability of annotation tools for both languages under consideration, while the alignment approach was developed with the assumption that annotation tools would only be available for one of the two languages. In principle the sentence-length filter could have been deployed, but it was seen in Chapter 2 that the sentence-length filter did not yield satisfactory results and we therefore opted not to deploy the filter at all.

On the influence of the filter on the results of the alignment approach when tools are available for both languages one can speculate that the filter can be of added value. It can be expected that applying the filter on the data before running the tools of the alignment approach will mostly have an effect on the quality of the alignments. The result will be that zero-alignments, i.e. words that do not get aligned to a word in the other language, and noisy alignment crossings will be less frequent, because the sentence pairs are more translationally equivalent and syntactically comparable. In general it can be expected that it will lead to more interpretable output of the tools and better hypotheses, however I did not experiment with the application of the filter to the alignment approach.

All in all, applying the filter is a trade-off between more comparable and

"cleaner" data and more interpretable output of the tools on the one hand, and the undesired removal of variation from the data on the other, which ties in with the discussion on what syntactic comparability is in Chapter 2. All of this raises the question: is using the filter for syntactically incomparable parallel sentences necessary when automatically detecting syntactic differences? I would like to hypothesize here that it depends on the sensitivity to noise of the method to detect syntactic differences that is used. The MDL approach fundamentally uses high frequencies in both the mining for patterns and the detection of differences, so it can be expected that the effect of the filter remains minimal as long as the size of the data is sufficiently large for the signal-to-noise ratio to be largely in favour of the signal – for the more frequent patterns, that is. As for the bottom half of the pattern lists, it can be expected that the effect of the filter is much larger, because a small change in frequency of a less frequent pattern (as a result of the filter) has a larger impact on its ability to efficiently compress the data and its statistical significance. The alignment approach, on the other hand, is probably much more sensitive to the effect of the filter, as was already discussed above.

In Chapter 3 it was already concluded that filtering out syntactically incomparable sentences is beneficial to the results. However, it depends on the situation, and the user should consider several things.

First, applying the filter drastically reduces the size of the data. When a user only has a fairly small dataset at their disposal, applying the filter may therefore be ill-advised. Though, when a user has a large dataset at their disposal, applying the filter may not be necessary when the tool used for the detection of syntactic differences is not very sensitive to noise, as was seen with the MDL approach, and may even be advised against due to the filter's computational expense, especially that of the GED-based filter. Applying the filter is therefore most interesting for middle-sized datasets, however it is very difficult to demarcate the boundaries of what constitutes a small, middle or large dataset. The issue of drastic data reduction would be greatly counteracted if a filter is developed that selects syntactically comparable sentence fragments. A possible way to achieve this is by for instance using punctuation to delineate smaller clauses and use those instead of full sentences, however the details to the implementation of this is left to future research endeavours.

Secondly, it depends on the noisiness of the data. As long as the signal-to-noise ratio is in favour of the signal, that is to say the data are clean, then applying the filter will not be necessary. However, when the data are noticeably noisy, i.e. containing many wrongly aligned sentence pairs, many free translation or syntactically incomparable constructions, then the user may opt to deploy the filter. It may therefore be advised first to run the MDL or alignment approach and to see if results are good.

Lastly, deploying the filter depends on the availability of a training set of sentence pairs, binarily labelled for syntactic comparability. The best filter was the combination filter, which was built on a logistic regression model and can only be used when a training set exists. Otherwise, the user would have

to resort to one of the other three developed filters and use a manually set threshold value, which may not be appropriate for the dataset in question.

As for the choice of which filter to use, the use of the combination filter is to be advised, but this can only be done, as said, when a training set exists. If such a set does not exist, the GED-based filter would be advised, but requires that there exist parsers for both languages that use the same annotation guidelines (such as Universal Dependencies). However, parses are rarely perfect which can lead to sentence pairs incorrectly being discarded, and, as mentioned, the GED-based filter is notably slow. The Levenshtein-distance filter is advised only when working with closely-related languages, because the Levenshtein distance is very sensitive to whole phrases transposing, and requires the existence of POS taggers for both languages that use the same annotation guidelines as well. If the user, for instance, is comparing English to Japanese, it would be ill-advised to use the Levenshtein-distance filter, but comparing Dutch to German should give reasonable results. The sentence-length filter is not advised, because it generally is too coarse-grained and does not use syntactic information.

5.3 Comparing MDL and alignment

In this Section I will compare the MDL approach of Chapter 3 to the alignment approach of Chapter 4. Very globally it can already be established that the MDL approach finds other types of differences than the alignment approach, simply because they process different types of data: the MDL approach uses linear POS tags and sequences, while the alignment approach operationalizes bitext word alignment and makes use of hierarchical dependency parses containing syntactic relations, POS tags and morphological features. Nevertheless, some valuable observations can be made when contrasting the results of the two approaches. Of course, the alignment approach put forth in Chapter 4 was developed from the assumption that no automatic annotation tools are available for one of the two languages under investigation, while the MDL approach of Chapter 3 requires the existence of (at least) POS taggers for both languages (that use the same tag set). Therefore the two approaches may be used in complementary situations, however for the purposes of this Section, I will assume a situation in which annotation tools are available for both languages so that both approaches could be deployed.

The foremost question is perhaps that of which type of syntactic differences can be found with the one approach but not with the other. The global answer to this question is that it depends on which information is passed to the system. As said above, the MDL approach uses POS tags and sequences, while the alignment approach uses dependency parses and alignment. The result is that any differences regarding syntactic function (i.e. dependency relation) or morphology can in principle not be found with the MDL approach without extensive manual research within the generated hypotheses, unless it is specifically coded into the POS tags. In Chapter 3 it was already discussed that the user could

opt for expanding the tag set such that it also reflects morphological, or indeed syntactic, information, by for instance appending the grammatical number to a POS tag. The issue with this, however, is that the MDL approach treats tags in a univariate way, i.e. a tag NOUN:Num=Sing (for singular nouns) is fundamentally distinct from a tag NOUN:Num=Plur (for plural nouns), as much as it is distinct from a tag for third person singular auxiliary verbs. This algorithmic behaviour is contrasted with the alignment approach, in which all annotation is processed in a multivariate way, such that the algorithm recognizes that a singular noun and a plural noun are both nouns and therefore more similar to each other than to an auxiliary verb. Expanding the tag set in MDL, therefore, is a trade-off between richer annotation and therefore more detailed syntactic differences that can be discovered on the one hand, and a loss of information and desired similarity between words due to further discretization of the data on the other.

Due to its more coarse-grained input and univariate nature, it can be concluded that the MDL approach is more prone to overgeneralization than the alignment approach. For instance, with the MDL approach it can be detected that pro-drop is extant in Finnish, but because the algorithm cannot distinguish between first, second and third person pronouns without creating more tags, causing the issue described above, it cannot directly show the linguist that Finnish pro-drop only affects non-third person pronouns.[1] A difference found with MDL should therefore very expressly lead to further investigation.

Meanwhile, the opposite holds true for the alignment approach. Its multivariate way of processing data and access to more detailed annotation lead it to being more prone to undergeneralization. This was for instance seen with the missed difference M4 from Chapter 4, which signified Hungarian's pro-drop also applying to singular object pronouns: I undergeneralized over the output of the tools and only concluded from the data that Hungarian has subject pro-drop. Although we have seen that the output of the alignment approach can lead to overgeneralization, too, the linguist may fail to detect a difference or feature as a result of being confronted with too much information.

Some smaller observations can also be made when comparing the two approaches. Related to the dropping of material, a notable difference between the two approaches is their applicability in tracking potential words or word types that are not overt in the other language, often involving functional material such as articles or personal pronouns which may be dropped or even be entirely absent in a language. Because the alignment approach operationalizes word alignment, it is fairly straightforward to track with it which POS tags (or even which combination of attributes of a word) often remain unaligned and untranslated in the unannotated target language: the developed tools retrieve the frequencies of unaligned cases of particular (combinations of) attributes, which in Chapter 4 quickly laid bare that Hungarian exhibits pro-drop, be-

[1] Only in very specific cases can third person pronouns be dropped in Finnish, such as answers to yes-no questions or when the dropped pronouns is c-commanded by a pronoun that is spelled out (Holmberg 2016).

cause pronouns remained unaligned very often. The MDL approach, however, can (and did) also detect that, e.g., pro-drop is extant in a language, although it is less straightforward to do so. In Chapter 3 it was shown that patterns with a pronoun in it were often absent in Czech while they were present in English and Dutch, strongly suggesting there may be pro-drop in Czech, but this must be deduced from the ranking of the patterns that contain a pronoun tag.

Another example that was already addressed in Chapter 3 and which may cause a linguist to miss that a word or word type is absent in one of the languages under investigation, is that of Ancient Greek and Turkish: whereas Ancient Greek only has definite articles, Turkish only has indefinite articles, which means that in every case that Ancient Greek has an article, Turkish will not have an article, and vice versa. Because definite and indefinite articles are tagged uniformly as DET in Universal Dependencies, and because the MDL approach does not use alignment to count the mismatches of patterns, the linguist may miss that articles exist in a complementary distribution in Ancient Greek and Turkish. The alignment approach is better at detecting this difference, due to it using word alignment and it having access to the subcategory attributes that distinguish definite from indefinite articles. It must however be noticed that the alignment method was designed to work on a language pair in which one of the two languages does not have available annotation tools, and due to the unilateral mapping of linguistic annotation from the source language onto the target language based on word alignments, the user may fail to detect any morpho-syntactic features that concern unaligned words in the target language. For example, let us assume that there are no annotation tools available for Ancient Greek, then the fact that indefinite articles are absent in Ancient Greek can be detected due to the Turkish indefinite articles remaining unaligned, however, because definite articles are absent in Turkish, no linguistic annotations are mapped onto the Ancient Greek definite articles through alignment, leading to the Ancient Greek definite articles being completely absent in the output of the developed tools.

Furthermore, the MDL approach is better at detecting differences in the linear ordering and adjacencies of elements. While the alignment approach does take into consideration the relative order by counting crossing alignments, it only shows the linguist, e.g., that an auxiliary verb comes before the main verb in Dutch (in main clauses), but it shows only very indirectly that there may be interfering material, such as adverbials or an object. The linguist may therefore miss the difference with English, where the possibility of intervening material between the auxiliary and the main verb is highly restricted. Related to this weakness of the alignment approach is that it was missed in Chapter 4 that Hungarian demonstrative pronouns must be directly followed by a definite pronoun. It was already suggested in that Chapter that the tools should consider adjacencies of words, so that these types of collocations in the target language can be discovered, however this will likely not solve the issue with interfering material.

The alignment approach has the advantage over the MDL approach that

it also considers word forms, which makes it possible to deploy the developed AAA tool, designed to detect potential affixes in the target language and to associate them to attributes of the annotated source language. It also makes it possible to detect morphological properties from the output of the other tools, for instance in Chapter 4 the fact that Hungarian has grammatical case.

All these discussed differences follow from the difference between the two approaches in information input and the way in which it is processed. An interesting line for future research would be to adapt the MDL approach to process syntactic trees in a multivariate way. Instead of linear POS sequences, it would then extract patterns that are parts of syntactic trees, in which nodes (i.e. words) contain multiple channels of annotation, with the preferable possibility of gapping over words akin to SQS, although it would require the existence of parsers for both languages, making it less broadly deployable. It is currently also unclear how this could be implemented and whether it would be computationally feasible.

It may furthermore be valuable to briefly discuss the difference in complexity of the outputs. While allowing for gaps in the patterns intuitively makes it easier to map differences in e.g. the use of articles, it was observed that gaps can make interpretation very complicated. Because the SQS algorithm used allows that the number of elements skipped over be strictly one less than the length of the pattern under consideration, it becomes increasingly difficult to understand a pattern as it grows in length. A pattern consisting of nine tags, such as PUNCT DET NOUN AUX ADP NUM NOUN VERB PUNCT, found in the English-Dutch run in Chapter 3, may have skipped over eight other tags, such as an adjective, an adverb or a verb, making it hard for the linguist to translate this sequence into something meaningful from which to derive a hypothesis on syntactic differences.

Apart from the difficulties that may arise from gapping, the MDL output is much more straightforward than the output of the alignment approach. Whereas the MDL approach ranks its output on relevance, the alignment approach does not, leaving the linguist to fully interpret the data by themselves, which may demand more practice.

So, from the point of view of the user and the usability of the tools, the choice between the MDL approach and the alignment approach is a trade-off between richer, more detailed annotation and therefore more detailed differences found on the one hand, and a much more complex interpretation of the output of the tools on the other. MDL more easily guides the linguist where to investigate further, whereas the alignment approach requires more input from the linguist to generalize and to find directions for further investigation.

5.4 General observations and findings

Over the entirety of this dissertation, some more general observations were made. In this Section I will discuss several findings that come to light when

comparing all three Chapters together.

5.4.1 On tagging and automatic annotation

An important observation that was made concerns the quality of tagging and parsing. The tools from all Chapters were shown to be very sensitive to tagging accuracy and consistency. As mentioned, a tagging error may push a sentence pair over the threshold and cause the filters from Chapter 2 to discard it wrongly. This is because a tagging error constitutes a higher edit distance, and because the edit distance is a discrete integer value, there is not much room for small errors.[2]

Chapter 3, too, saw a strong influence of tagging errors on the results, because of a ripple effect down the line. A tagging error causes a distortion in the frequency of a pattern, causing it to compress the data less well and reducing the chance for it to be mined by SQS. A distortion in the frequency of a pattern due to a tagging error also distorts the frequencies of the mismatches, which are crucial in the ranking of the differences, and may cause the difference to be ranked much lower than it should have been, and to be missed by the linguist.

Tagging errors also cause issues for the tools of Chapter 4. Because the alignment approach uses so much annotation – not only POS tags – the chances of one of the attributes to be incorrect goes up. This causes the output to be very noisy, which may cause syntactic differences to be missed, partly because the noisiness raises the necessity for suppressing the output.

Tagging inconsistencies, as opposed to tagging errors, also raised issues in a similar way for the filters and the MDL approach. Whereas a tagging error is the assigning of a wrong label, a tagging inconsistency is the assigning of a label that is justified within the grammar of a language, but not between two languages. If the two languages under investigation have even slightly different annotation guidelines, a NOUN tag in the one language may not fully correspond to a NOUN tag in the other, which will lead to more mismatching occurrences and consequently to patterns with a high χ^2 value that in fact do not indicate a syntactic difference. As pointed out in Chapter 3, we found that in English many more words were tagged as PROPN than in Dutch and Czech, despite having clear nominal or adjectival morpho-syntactic properties and the direct translations in the latter two languages were often tagged as nouns or adjectives, capitalized or not. Although it may be true and solidly justified to have the words be tagged as proper nouns in a language's linguistic tradition, this inconsistency led to the MDL approach finding many syntactic differences between English and the other two languages that arguably do not signify true differences in the syntactic potential of the languages in question. While it was observed that Universal Dependencies guidelines may not always

[2] Of course, this does not hold true for the sentence-length filter, because it does not use syntactic information. A tagging inaccuracy therefore has no effect on its results.

be as consistent throughout languages as desired, the contribution that Universal Dependencies have made to the universalization of annotation guidelines throughout languages and therefore the possibility to more efficiently compare languages to one another cannot be denied and has proven vital in this dissertation and beyond.

5.4.2 Corpus choice

On the matter of corpus genre, it was observed that both the Europarl corpus and the Bible, used throughout this dissertation, were rather particular in their language use. The Europarl corpus shows a very high average sentence length and frequent formulaic utterances common for language used in Parliament, and the Bible shows many archaisms, distinguishing both corpora from day-to-day language. The result is that certain constructions are overrepresented in the data while others are underrepresented. Despite their shown effectiveness in the detection of syntactic differences, the tools developed in Chapters 3 and 4 were therefore not able to detect every difference between the languages under investigation. Of course, corpus choice and the genre of the corpus are crucial in any natural-language processing task, as was also pointed out by Wälchli (2007), which was extensively discussed in Chapter 1. As a result, one of the conclusions of this dissertation is that corpus choice influences the results of the automatic detection of syntactic differences, and that a potential user of the tools must be aware of the possibility of syntactic differences being missed.

As for corpus size, it is difficult to say how large a dataset should be in order to be able to successfully detect syntactic differences automatically from it. Chapter 4 generally describes good results, although some characteristic differences between English and Hungarian were not found, but the Bible, with a version containing 28,972 verses used in this dissertation,[3] is considered to be a relatively small corpus and one could expect to be able to detect the missed differences using a larger corpus. However, the data used in Chapter 3 were much smaller (only 10,000 sentence pairs)[4] – especially after filtering out syntactically incomparable sentence pairs which saw a reduction of the data to one fifth to one sixth of the original number of sentence pairs – and good results and meaningful hypotheses were nonetheless obtained. The influence of corpus size was all in all not strongly noticed: the reduction of corpus size due to the filter only marginally influenced the results, and no differences between the MDL approach and the alignment approach could be traced back to a difference in corpus size. This is in line with Sanders (2007), who showed that the size of the data can be reduced in comparison to Nerbonne and Wiersma (2006), and can in fact be relatively small in order to be able still obtain significant results. Sanders (2007) argued that there may be a lower limit to the data size of around 250,000 words (for his method, at least). However, during the MDL

[3] Containing around 850,000 tokens for English and 680,000 for Hungarian.
[4] Containing around 220,000 tokens for English, 225,000 for Dutch and 190,000 for Czech.

experiments with the filter many fewer words were used (between 9,000 and 17,000, depending on the language pair), but results were still significant.

In fact, using very large corpora may not be advisable. This is not only because it may make the interpretation of the results of the MDL approach and especially the alignment approach even more complicated, but mostly because the algorithms of the filter, the MDL approach and the alignment approach are computationally complex. As for the filter, especially the GED-based filter is computationally complex, given that it was proven that calculating the exact GED is NP-hard (Zeng et al. 2009) and that the problem is even APX, meaning that it is hard to approximate as well (Lin 1994). The MDL approach is computationally expensive due to its relying on the SQS algorithm, the complexity of which can grow cubically with the size of the data, although in practice it is much faster (Tatti and Vreeken 2012). Finally, the alignment approach also suffers from data size limitations, especially the GTI, which produces a massive output as a result of iterative nesting, and the AAA, which has a looming danger of combinatorial complexity (a growth curve even worse than exponential). There may therefore be an upper limit to the size of the data that can be used with the tools developed for the purposes of this dissertation, however it is hard to determine this limit.[5]

The use of parallel corpora was shown to be of added value to the automatic detection of syntactic differences. Although Wiersma, Nerbonne and Lauttamus (2011) already successfully extracted syntactic differences from non-parallel corpora, the use of parallel corpora allowed us to identify in which contexts the differences occur, and even to generate hypotheses on syntactic differences between an annotated language and an unannotated language with the help of alignment (which is only possible in parallel corpora). The MDL approach, the way it is designed in this dissertation, also relies on parallel corpora, because it counts the mismatches of patterns between sentence pairs, which allows for more precise frequencies and circumvents the need for a complex statistical test to mitigate for non-parallellity – although an adaptation to the algorithm could probably be devised so that it works on non-parallel data, too.

Because of the way the tools were designed, I did not compare results from experiments with parallel data with results from experiments with non-parallel data, although differences with the results from others were discussed in the previous Chapters (chiefly among which Nerbonne and Wiersma 2006; Wiersma, Nerbonne and Lauttamus 2011). Wälchli (2007) already extensively argued for the use of parallel corpora, as discussed in Chapter 1. To add to this discussion, it is most desirable to use very homogenous data when trying to detect syntactic differences between languages, so that any variation found between

[5] I also firmly believe that the complexity of the algorithms and the size of the data should be considered more often in academia, because the carbon footprint of complex calculations is much higher than people realize. My colleague dr. Alex Brandsen already noted that the carbon footprint from the GPU usage during his PhD research was equivalent to that of a flight from Amsterdam to Prague, and that less computationally expensive methods are therefore preferable (Brandsen 2022: proposition no. 7).

the languages can be traced back to the syntactic variation. The use of a parallel corpus removes unwanted sources of variation, such as variation in speaker, genre, and text length, making it ideal for the purposes of comparative-syntactic research.

One point of concern regarding the use of corpora (at all, both parallel and non-parallel) is that it has a confirmation bias, because in general they only contain correct utterances, while in comparative-syntactic research it can be very insightful to have a few ungrammatical sentences at one's disposal,[6] especially when access to large datasets is limited: the range and limits of syntactic variation are not merely defined by what can be said, but also by what cannot be said. The tools developed for and presented in this dissertation should therefore always be considered as complementary to traditional comparative-syntactic research.

5.4.3 Some remarks on future research

In the task of automatically detecting morpho-syntactic differences between languages, it is important that the output of the algorithm, as well as the algorithm itself, are transparent and interpretable for the human linguist, so that phenomena can be researched more closely, cross-linguistic theories on syntactic variation can more easily be formulated and the research remains replicable and reproducible. While the interpretability of the algorithm and its transparency are known problems for deep learning approaches, the future may hold more direct applications of deep learning in the task of automatically detecting syntactic differences,[7] especially in light of the more recent developments concerning the opening of the 'black boxes' that deep learning models are famous for. The architecture of a more deep-learning driven approach to detecting syntactic differences, though, remains unclear. Ideally a transparent and interpretable unsupervised deep-learning method will be deployed, in which the output is not restricted to predefined labels and syntactic differences can be detected that were hitherto unknown.

A more clear future for machine learning approaches can be seen when labels for morpho-syntactic properties of languages or language varieties are already available, in which case the properties can be used to cluster languages based on syntactic 'behaviour' so as to cluster languages on their phylogenetic relationship (cf. Spruit 2008, who clustered Dutch dialects based on discrete syntactic properties), or to detect associations and correlations between the properties so as to reduce them to fewer overarching syntactic properties or phenomena (cf. e.g. also Spruit 2008, as well as Van Craenenbroeck, Koppen

[6] These ungrammatical sentences have usually been very carefully selected or in fact, in most cases, been constructed.

[7] This dissertation already saw the use of deep learning methods in less direct ways, namely in the preparation of the data. UDPipe, for instance, uses models that are trained using deep learning algorithms, but the transparency of the tools for data preparation were deemed to be of less importance than of the algorithms that detect the differences.

and Bosch 2019).

As for the future of the influence of the human linguist in the process of automatically detecting syntactic differences, I think it can be stated that the human linguist cannot be removed from the equation. As already said in Chapter 4, I believe that a good balance can be struck between the freedom for subjective interpretation on the one hand and the more computer-driven generation of hypotheses on the other, though whatever the tendency in the balance struck, the expertise and subjective interpretation of the linguist will always be there: either the linguistic bias will be present in the interpretation of the output, or the linguistic bias will be put in the design of the algorithm.

The question of what this balance should look like is interesting, however. In Chapter 3 and 4 it was already seen that the approaches require drastically different inputs from the linguist: while the difficulty with MDL mostly resided in the interpretation of the longer patterns and specifying (as opposed to generalizing over) differences by going back to the data, the difficulty with the alignment approach mostly resided in making sense of zero-alignment frequencies, crossings and other annotations and generalizing over several differences that cover one larger phenomenon. The latter of the approaches required more practice, and in the future the linguist could definitely benefit from a better user interface. It would even be possible to have the linguist interact with the algorithm during the process.

5.5 Conclusion

Relating this all back to the research question of whether it is possible to automatically detect syntactic differences and, if so, how, it was shown that correct hypotheses on syntactic differences between languages can be generated from parallel corpora through the use of the minimum description length principle, counting mismatches between part-of-speech pattern occurrences, word alignment and mapping annotation from an annotated language onto another unannotated language. The automatic detection of syntactic differences between languages is therefore possible, yes. The tools developed for the purposes of this research work well and can aid a linguist significantly in their search for differences or similarities. However, it was also seen that the tools do not work perfectly, for instance hampered by the quality of the data and annotations, and the process may, for now, not be as detailed, automatized or objective as one would wish, leaving much room for future endeavours.