Is Old French tougher to parse?

Loïc Grobol[1, 2], Sophie Prévost[3], Benoît Crabbé[4]
(1) Modyco, Université Paris Nanterre et CNRS
(2) LIFO, Université d'Orléans and INSA Centre – Val-de-Loire
(3) Lattice, CNRS, ENS, PSL and Université Sorbonne Nouvelle
(4) LLF, CNRS and Université de Paris
sophie.prevost@ens.psl.eu, lgrobol@parisnanterre.fr, benoit.crabbe@linguist.univ-paris-diderot.fr

Abstract

Medieval French is known to be relatively hard to parse, with several possible sources of confusion for automatic parsers, among which its flexible word order and its graphical and syntactic variation, both synchronically and diachronically. In this work, we study in particular the influence of word order, by comparing the performances of two state-of-the-art syntactic parsers trained and evaluated on two treebanks: the Syntactic Reference Corpus of Medieval French (SRCMF), a treebank of Old French (9th—13th century) and the Google Stanford Dependency treebank of contemporary French.

1 Introduction

Parsing Old French is thought to be hard because the language has flexible word order, graphical and syntactic variation. As a result, automatic parsers are underperforming for Old French as compared with most other Romance languages when accounting to the amount of available data (Zeman et al., 2018).

However, while previous studies (Stein, 2014; Stein, 2016; Guibon et al., 2014; Guibon et al., 2015) have investigated the issue with parsing from an intrinsic point of view, to our knowledge, there is no comparative study of the impact of these characteristics on the behaviour of automatic parsers. In particular, there has been no specific study attempting to assess the impact of Old French free word order on parsing.

In this work, we propose a first step in these directions by studying automatic dependency parsing of Old French as compared to Contemporary French. To this end, we train state-of-the art parsers on the closest alter ego in both languages: the Syntactic Reference Corpus of Medieval French (UD-Old-French-SRCMF, henceforth SRCMF), a treebank of Old French (9th—13th century) and the Google Stanford Dependency treebank of contemporary French (UD-French-GSD, henceforth GSD) ; both from the Universal Dependencies (Nivre et al., 2020) projet.

Both corpora have some similarities —comparable sizes and French language— and some dissimilarities as they represent different stages of the French language, with noticeable linguistic differences between them. Our aim is to assess whether those discrepancies have an impact on the scores of the parser and on the types of errors that they make.

We propose a quantitative and qualitative error analysis with a particular focus on word order, with the following intuitive hypothesis: considering the flexibility of word order as well as the morphological variation in Old French, we expect lower scores on the SRCMF treebank than on the GSD treebank and different types of errors.

2 Data

The Universal Dependencies Old French Syntactic Reference Corpus for Medieval French (SRCMF) treebank (Stein and Prévost, 2013) consists of texts spanning from the 9th to the 13th century. In its

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: http://creativecommons.org/licenses/by/4.0/.

most recent version, it consists of 199 700 tokens (including punctuation marks) in 18 030 sentences for an average of 11.1 tokens per sentence.

Most of the development and test data is taken from texts sharing properties with training data, but conversely pre-1100 texts only appear in the training set because they were deemed too small to reserve anything for testing.

The Universal Dependencies French Google Stanford Dependency (GSD) treebank (Guillaume et al., 2019) consists of Contemporary French data, mainly from encyclopedic articles and tourist reviews. It includes 400 399 tokens for 16 341 sentences (averaging 24.5 tokens per sentence). There is no broad chronological span, but the genre disparities may entail significant internal variability. The split sizes for both corpora are reported in table 1.

	Train		**Dev**		**Test**	
Corpus	Tokens	Sentences	Tokens	Sentences	Tokens	Sentences
SRCMF	158 620	14 153	20 554	1888	20 526	1989
GSD	354 662	14 449	35 718	1476	10 019	416

Table 1: Corpus sizes, using their respective standard splits in Universal Dependencies

There are two explanations for the wide gap between the sentence lengths in the two corpora — which might influence the performances of the parser. The first explanation is a linguistic one, as sentences in Contemporary French tend to be more complex (and thus longer) than in Old French, especially because they include more subordinate clauses. The second one is methodological, and lies in a different representation of coordinated clauses: in SRCMF, any finite verb of a main clause gives rise to a sentence and there is no coordination between main clauses. On the contrary, in GSD, main clauses may be coordinated in a single sentence under specific conditions (if the second verb has no overt subject). Hence, the following example is analysed as a single sentence in GSD while it would analysed as two separate sentences in SRCMF: "*Selon Alan «Dave m'a contacté il y a quelques semaines et m'a demandé si je serais prêt à les rejoindre sur scène»*" ("Dave contacted me a few weeks ago and asked if I would accept to join them on stage").

These differences in sizes could have an influence on the global performances of learned parsers, however neither the direction nor the magnitude of the difference is clear from the current state of the art. Grobol and Crabbé (2021) for instance report better performances on the Sequoia treebank (Candito and Seddah, 2012) than on GSD, despite its smaller size, but worse performances on the French Treebank (Candito et al., 2010), which is larger than GSD. However, since our analyses in this work focus on tree-level behaviours rather than word-level performances, our hypothesis will be that since GSD and SRCMF have a similar number of trees, it makes sense to compare parsers trained on these treebanks.

3 Parser

The parser used in this study is HOPS (Grobol and Crabbé, 2021), a neural graph parser/POS tagger with state-of-the-art results on the Universal Dependencies contemporary French corpora. More specifically, HOPS is variant of Dozat and Manning (2018)'s Biaffine graph parser, that takes transformer language model representations as inputs and where POS-tagging is not an explicit step independent of parsing, but it is instead performed jointly with parsing in a hard parameter sharing (Ruder, 2017) multitask formulation as in e.g. Coavoux and Crabbé (2017). Beyond its sheer performances, our choice was also motivated by the versatility of this parser regarding word representations, as it is able to simultaneously use contextual and non-contextual word embeddings along with character-level representations, which — as noted by Smith et al. (2018) — can have a significant impact for parsing languages with rich morphologies and/or flexible graphic systems. In all our experiments, we used the same hyperparameters as Grobol and Crabbé (2021) for their FlauBERT-based models.

In order to parse UD-French-GSD, we retrain a parsing model from scratch, using a French transformer model, FlauBERT-base (Le et al., 2020), for the contextual word embeddings. The results in terms of POS

tagging, unlabelled attachment and labelled attachment F-scores are reported in table 2 and are similar to those reported by Grobol and Crabbé (2021).

Partition	UPOS	UAS	LAS
Dev	98.63	96.71	95.60
Test	98.61	95.90	94.35

Table 2: Performances of the parser (dev-best model out of 5 random seeds) on GSD development and test partitions.

Grobol and Crabbé (2021) show that using Transformer-based contextual word embeddings (Devlin et al. (2019) among others) greatly improves dependency parsing for contemporary French. In order to benefit from comparable advantages when parsing Old French, we derive adapted contextual embeddings in two different ways: by training a small RoBERTa model from scratch (Micheli et al., 2020) and by further training of FlauBERT (Le et al., 2020), in both cases on a corpus of raw Old French and early Middle French of about 10 Mwords[1]. This results in a situation where despite the resources disparities between Old and Contemporary French in general, the parsers have access to comparable resources for both languages.

Development	UPOS	UAS	LAS
HOPS (scratch)	97.14	92.95	89.18
HOPS (FlauBERT)	97.72	93.70	90.93
Test	**UPOS**	**UAS**	**LAS**
Straka et al. (2019)	96.26	91.83	86.75
HOPS (scratch)	96.60	92.20	87.95
HOPS (FlauBERT)	97.59	93.73	90.98

Table 3: Performances of the parser (dev-best model out of 5 random seeds) on SRCMF development and test partitions.

Table 3 reports the results obtained using these two strategies to obtain contextual word embedding. We note that our parser obtains rather good scores and improves on the state-of-the-art with a considerable margin (which is not very surprising, since unlike Straka et al. (2019) we could rely on specific monolingual contextual word embeddings). Considering these results, the rest of our analyses focus on the better-performing FlauBERT-based model. To preserve the opacity of the test partition and avoid design overfitting (van der Goot, 2021), we will only consider the development set of both corpora in the rest of this work.

4 Comparative analysis

As interesting as UPOS, UAS, LAS may be from a computational point of view, when using automatic parsing as a preprocessing step for a large-scale linguistic analysis, the proportion of trees that are fully correctly parsed is also relevant.

For SRCMF, a total of 1100 sentences (58.26 %) of the sentences are parsed completely correctly and 788 sentences (41.74 %) have at least one parsing error (either a wrong attachment or a wrong dependency label. For GSD, we find 660 sentences (44.72 %) of completely correct parses.

Therefore, somewhat unexpectedly, the parser obtains better results on SRCMF than on GSD for these metrics. However, the picture is different with a more refined analysis. If we focus on the major constituents, that is Subject, Object, Root and Copula, the picture is quite different. There are 1693 sentences

[1]This corpus consists of texts from the BFM (Guillot et al., 2018), AND (Trotter, 2012), NCA (Kunstmann and Stein, 2008), Chartes Douai (Glessgen et al., 2010), OpenMedFr (Wrisley, 2018), Geste (Camps et al., 2019), MCVF (Martineau, 2008) and Chartes Aube (Van Reenen et al., 2006) corpora.

(88.67 %) in SRCMF where there are no errors on these syntactic functions, whereas in GSD this amounts to 1423 sentences (96.21 %).

These results show that the parser is better for non major constituents (such as adverbial phrases or clauses, or internal structure of NPs) on Old French, and for major constituents in contemporary French.

Two complementary observations could explain this difference. First, Old French is characterised by high variations. A word can be spelled in many different ways, null subjects are very frequent, and word order has a considerable flexibility, allowing for preverbal objects, postverbal subjects, and all six permutations of S, V and O (SVO, SOV, OVS, OSV, VSO and VOS), even though SVO is the prevalent form very early on. This multi-dimensional variation probably makes it more complex for the parser to correctly identify subjects, objects (or even verbs) in SRCMF than in GSD. Secondly, as mentioned above, sentences are longer and more complex in GSD than in SRCMF, with either more peripheral elements and/or more complex NPs, which may be difficult to correctly analyse for the parser.

From now on, we will focus on two major constituents, subjects and objects, —both nominal or pronominal— in main declaratives. We thus examine cases of parsing errors of the nsubj and obj relations[2], while taking into account the respective order of the main constituents in the sentence.

We focus first on the orders which are common to both treebanks, in order to determine whether there are, for instance, more errors on the attachment and/or the label of the object in OVS than in SVO in either treebank. Then we will turn more specifically to SRCMF, taking also into account unattested combinations in, trying to highlight some correlations between wrong parsing and the different combinations. One important concern, as historical linguists, is not to miss alternative orders to SVO: even though they are much rarer, they constitute a major feature, and their decrease represents a very important evolution in the history of French. Table 4 reports the error rates for subject and object relations according to the different word orders in both corpora

	SRCMF			**GSD**		
Order	sentences	nsubj errs (%)	obj errs (%)	sentences	nsubj errs (%)	obj errs (%)
V	472	2.12	5.72	105	1.90	0.00
SV	424	4.25	1.65	895	0.78	1.67
VS	173	6.94	2.89	26	3.85	0.00
SVO	119	4.20	16.81	384	0.52	2.34
SOV	109	5.50	5.50	52	0.00	11.54
OVS	71	14.08	15.49	1	0.00	0.00
VO	250	4.80	8.00	13	0.00	0.00
Total	1618	4.80	5.93	1476	0.81	2.24

Table 4: Comparison between the error rates for the nsubj and obj relations in both corpora and in the common word orders.

Table 4 shows that i) in both corpora the parser tends to be more high-performing for nsubj than for obj, with exceptions for the SV, VS and V orders; ii) the parser always performs better on Modern French than on Old French (except for obj in SV, but the difference is insignificant : 1,67 vs 1,65). We now turn to a qualitative analysis of the errors that both corpora have in common: wrong nsubj in V, SV, VS and SVO and wrong obj in SV, SVO and SOV.

4.1 Errors on subjects

V order in GSD expl:subj (PRO *ce*) are parsed as nsubj twice. In SRCMF, we find three main types of errors: either a preverbal oblique is parsed as a nsubj, nsubj is wrongly attached to the root, or nsubj is correctly attached to a wrong root.

[2]We restrict our study to nouns and pronouns, since clausal constituents obey different mechanics. Therefore, we leave csubj and ccomp aside in this work.

SV order in GSD there are two main types of errors: in a complex NP a dependent element is labelled as the head, i.e. as nsubj; nsubj is attached to a wrong root. In SRCMF, there are three types of errors: most often, nsubj is wrongly labelled (as xcomp, root, obj, vocative, csubj, nmod) which results in the absence of any nsubj; in a few cases, an element is wrongly labelled as a nsubj, which results in a double nsubj; in another few cases, nsubj is attached to a wrong root.

VS order in GSD nsubj is wrongly labelled only once, as an xcomp while an apposition of an obl is labelled as nsubj. In SRCMF, nsubj is wrongly labelled as an obj or an obl in half cases, but also as a root or a case, or attached to a wrong root. In most cases this entails the absence of nsubj; an amod is once wrongly labelled as an nsubj, which results in a double nsubj.

SVO order in GSD there are two errors: nsubj is labelled as flat (and flat as nsubj) or nsubj is attached to a wrong root. In SRCMF, either nsubj is labelled as an obl or an obj or a root, or it is attached to a wrong root.

SOV, OVS and VO orders nsubj are all correctly parsed in GSD. In SRCMF, in SOV and OVS, most often nsubj is wrongly labelled as obj (which sometimes entails a double obj), obl, xcomp, disloc, apposition; sometimes it is attached to a wrong root; exceptionally an obj is labelled as a nsubj (leading to a double nsubj). In VO, on the contrary, the most frequent error is the wrong labelling of a category (mainly obj) as a nsubj (with cases of double nsubj).

4.2 Errors on objects

SV order in GSD, most errors consist in the wrong analysis as an obj of *se* (PRO), which is expected to be an expl:pass. In other cases, an obl or xcomp is wrongly labelled as an obj. In SRCMF, a nsubj or xcomp is wrongly labelled as an obj, or obj is attached to a wrong root.

SVO order in GSD most errors result from the analysis of obj as xcomp in existential contructions such as "*cette disparition reste une énigme"* ("this disappearance remains a puzzle" (let it be noticed that the analysis as obj is not uncontroversial, as one could have expected to find an obl instead). In a few cases obj is wrongly parsed as obl. In SRCMF errors are far more diversified. Most often, obj is wrongly analysed (obl, advmod, amod, advcl, root, nsubj, hence a double nsubj in an unexpected order SVS) ; in a few cases, obj is wrongly attached.

SOV order in GSD, obj can only be a pronoun. Most errors result from the analysis of reflexive *se* as an expl: pass instead of an obj, both analyses being actually acceptable. In SRCMF, we find nominal objects (albeit rarely: "*Li rois Tristran menace"*). In some cases, obj (nominal or pronominal) is wrongly parsed (obl or flat), or a category is wrongly parsed as an obj, in addition to the correct obj (hence a double obj).

V, VS, OVS and VO orders obj are all correctly parsed in GSD. In SRCMF errors in V and VS orders necessarily involve a category being wrongly parsed as an obj. In V order, in most cases, an obl is wrongly analysed as an obj in existential constructions (where the SRCMF scheme expects obl). In VS, most often nsubj is wrongly parsed as an obj. In VO, most often obj is wrongly analysed (nsubj, obl, nmod, flat, ccomp). It is rarely wrongly attached and the root is correct in most cases. The same holds true for OVS (obj wrongly parsed as obl, advmod, iobj, root, nsubj, hence a double nsubj), though we also find one nsubj parsed as obj, hence a double obj.

Finally, to summarise these analyses, we can note a few main trends:

- both treebanks display both types of errors for nsubj and obj: the absence of a correct label (recall) and/or the presence of a wrong label (precision);
- in GSD a wrong label is never correlated to a wrong part-of-speech, whereas this is the case in 10 % of cases (16/169) in SRCMF;
- not only are the scores better in GSD than in SRCMF, but the errors are usually of a different nature, and less damaging: wrong parsing of obj is always at the benefit of a close category (obl, xcomp,

expl: pass), and this also holds true for nsubj (flat, appos, amod, expl:subj). Wrong attachments or wrong roots are exceptional[3]. On the contrary, in SRCMF, wrong roots and wrong attachments are not an exception[4], and wrong parsing of nsubj and/or obj often results in distant categories, with even possible inversions between nsubj and obj.

5 Influence of word order frequencies for parsing SRCMF

We now turn more specifically to SRCMF, in order to highlight a few correlations between frequencies of word orders and performances of the parsing. Table 5 reports the error rates for the nsubj, obj and root relations in all eleven attested combinations.

	Prevalence		**Error rates** (%)				
Order	sentences	%	root	nsubj	obj	core	any
V	472	25.02	2.75	2.12	5.72	9.32	40.04
SV	424	22.48	2.36	4.25	1.65	6.60	36.79
VS	173	9.17	2.89	6.94	2.89	8.67	36.42
SVO	119	6.30	4.20	4.20	16.81	16.81	51.26
SOV	109	5.77	0.92	5.50	5.50	7.34	42.20
OVS	71	3.76	1.41	14.08	15.59	19.72	53.52
OSV	17	0.90	0.00	5.88	0.00	5.88	47.06
VSO	23	1.21	0.00	0.00	30.43	30.43	69.57
VOS	7	0.37	14.29	14.29	0.00	28.57	71.43
VO	250	13.25	1.20	4.80	8.00	10.40	42.00
OV	221	11.71	0.00	2.71	6.79	6.79	44.80
Total	1888		2.12	4.34	6.30	9.54	41.74

Table 5: Comparison between the error rates for the root, nsubj and obj relations in SRCMF in all the occurring word orders. The "core" column is the ratio of trees where are at least one relation among root, nsubj and obj has an error.

From table 5, it appears that there is no significant correlation between word order frequency and parser performances: the five most frequent orders (V > SV > VO > OV > VS) respectively rank as 6, 3, 7, 2 and 5 in terms of total error rate. On the contrary SOV and OSV, ranked respectively 7 and 10 in terms of frequency, are respectively in position 4 and 1 position in terms correctness.

From a qualitative point of view, we may account for the high performance for SOV by the fact that obj is most often a pronoun (*le*, *la*, *les*, …), with an unambiguous function (versus an NP, which can be obj or nsubj), which probably reduces options and hence errors. For OSV, in most cases (14/17), either nsubj or obj (or both) is a unambiguous pronoun. It should also be noticed that the verb is always a complex one, hence a structure such as: obj aux nsubj verb.

Furthermore, orders with both nsubj and obj are the least frequent ones, and, globally, those show the worst parsing performances, which can be accounted for by the higher complexity of the tree. On the contrary, the high score of wrong parsing in VO is unexpected as i) VO is not infrequent (13.25 %, 3rd position) and ii) obj is often wrongly parsed as an nsubj, with results in a VS order, less frequent (9.17 %). Globally speaking, the highest rate of wrong parsing concerns obj (6.3 %), followed by nsubj (4.34 %) then by root (2.12 %). However this hierarchy varies according to word orders, at least as concerns nsubj and obj, since root always displays the lowest rate of errors (except in V order). Getting back to linguistic concerns, we observe that 3 out of the 4 very rare (less than 5 %) word orders are very badly parsed, with error rates over 20 %: OVS, VSO and VOS, which is of course more damaging for subsequent linguistic studies than when it happens with more frequent orders.

[3]Wrong parsing of roots in general in the seven orders amounts to 0.54 %.

[4]Wrong parsing of roots in general in the seven orders amounts to 2.35 %.

6 Conclusion

Our results suggest that parsing Old French is indeed harder than contemporary French, at least in the current state of existing treebanks (in terms of amount of text and heterogeneity). This is manifested both in word-level metrics and in terms of exact match for major constituents. In term of exact match for all words, our results favour Old French, but this metric is very likely to be influenced by the significantly smaller sentence lengths. Qualitative analyses concur with this trend: parsing errors seem less severe in general in GSD. Finally, rather surprisingly, while errors are not homogeneous among word orders, the most common word orders are not necessarily those that are best dealt with, although the least common word orders are those where there are the most errors.

These are provisional conclusions which deserve further investigations, especially in order to refine the correlations between word orders and wrong parsing, be it as regards the types of errors or the factors likely to be of influence. Orthogonally to these considerations, a broad study of the impact of treebank sizes and sentence lengths on parsers' behaviours could also be a useful complement of this work.

We reserve for future work transverse analyses with other facets such as time period and genre that we have abstracted over in this work. Going forward, being able to narrow down the sources of errors could help design parsers with better handling of rare phenomena, which would be crucial to support fine-grained quantitative linguistic analyses.

References

Jean-Baptiste Camps, Elena Albarran, and Alice Cochet. 2019. Geste: un corpus de chansons de geste, 2016-…, April.

Marie Candito and Djamé Seddah. 2012. Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Actes de la conférence conjointe JEP-TALN-RECITAL 2012*, volume 2, Grenoble, France, June. Association pour le Traitement Automatique des Langues.

Marie Candito, Benoît Crabbé, and Pascal Denis. 2010. Statistical French dependency parsing: treebank conversion and first results. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, pages 1840–1847, Valletta, Malta, May. European Language Resources Association.

Maximin Coavoux and Benoît Crabbé. 2017. Multilingual Lexicalized Constituency Parsing with Word-Level Auxiliary Tasks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 331–336, València, España, April. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2018. Simpler but more accurate semantic dependency parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 484–490, Melbourne, Australia, July. Association for Computational Linguistics.

Martin-Dietrich Glessgen, Dumitru Kihaï, and Paul Videsott. 2010. L'élaboration philologique et linguistique des Plus anciens documents linguistiques de la France. *Bibliothèque de l'École des chartes*, 168(1):5–5.

Loïc Grobol and Benoît Crabbé. 2021. Analyse en dépendances du français avec des plongements contextualisés. In *28e Conférence sur le Traitement Automatique des Langues Naturelles*, Lille, France, June. Association pour le Traitement Automatique des Langues.

Gaël Guibon, Isabelle Tellier, Mathieu Constant, Sophie Prévost, and Kim Gerdes. 2014. Parsing Poorly Standardized Language Dependency on Old French. In Verena Henrich, Erhard Hinrichs, Daniël de Kok, Petya Osenova, and Adam Przepiórkowski, editors, *Proceedings of the Thirteenth International Workshop on Treebanks and Linguistic Theories*, pages 51–61, Tübingen, Deutschland, December.

Gaël Guibon, Isabelle Tellier, Sophie Prévost, Mathieu Constant, and Kim Gerdes. 2015. Searching for Discriminative Metadata of Heterogenous Corpora. In Markus Dickinson, Erhard Hinrichs, Agnieszka Patejuk, and Adam Przepiórkowski, editors, *Proceedings of the Fourteenth International Workshop on Treebanks and Linguistic Theories*, pages 72–82, Warszawa, Polska, December.

Bruno Guillaume, Marie-Catherine de Marneffe, and Guy Perrier. 2019. Conversion et améliorations de corpus du français annotés en Universal Dependencies. *Traitement Automatique des Langues*, 60(2):71.

Céline Guillot, Serge Heiden, and Alexei Lavrentiev. 2018. Base de français médiéval : une base de référence de sources médiévales ouverte et libre au service de la communauté scientifique. *Diachroniques. Revue de Linguistique française diachronique*, (7):168.

Pierre Kunstmann and Achim Stein. 2008. Le Nouveau Corpus d'Amsterdam. *Corpus*, (7), November.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France, May. European Language Resources Association.

France Martineau. 2008. Un corpus pour l'analyse de la variation et du changement linguistique. *Corpus*, (7), November.

Vincent Micheli, Martin d'Hoffschmidt, and François Fleuret. 2020. On the importance of pre-training data volume for compact language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7853–7858, Online, November. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France, May. European Language Resources Association.

Sebastian Ruder. 2017. An Overview of Multi-Task Learning in Deep Neural Networks. *arXiv:1706.05098 [cs, stat]*, June.

Aaron Smith, Miryam de Lhoneux, Sara Stymne, and Joakim Nivre. 2018. An Investigation of the Interactions Between Pre-Trained Word Embeddings, Character Models and POS Tags in Dependency Parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2711–2720, Bruxelles, Belgique, October. Association for Computational Linguistics.

Achim Stein and Sophie Prévost. 2013. Syntactic annotation of medieval texts: the Syntactic Reference Corpus of Medieval French (SRCMF). In Paul Bennett, Martin Durrell, Silke Scheible, and Richard J. Whitt, editors, *New Methods in Historical Corpora*, Corpus Linguistics and International Perspectives on Language, pages 275–282, Manchester, UK, September. Gunter Narr Verlag.

Achim Stein. 2014. Parsing Heterogeneous Corpora with a Rich Dependency Grammar. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 2879–2886, Reykjavík, Ísland, May. European Language Resources Association.

Achim Stein. 2016. Old French Dependency Parsing: Results of Two Parsers Analysed from a Linguistic Point of View. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 707–713, Portorož, Slovenija, May. European Language Resources Association.

Milan Straka, Jana Straková, and Jan Hajič. 2019. Evaluating Contextualized Embeddings on 54 Languages in POS Tagging, Lemmatization and Dependency Parsing. *arXiv:1908.07448 [cs]*, August.

David Trotter. 2012. Bytes, Words, Texts: The Anglo-Norman Dictionary and its Text-Base. *Digital Medievalist*, 7(0), February.

Rob van der Goot. 2021. We Need to Talk About train-dev-test Splits. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4485–4494, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

Pieter Van Reenen, Evert Wattel, and Margôt van Mulken. 2006. *Chartes de Champagne en français conservées aux Archives de l'Aube, 1270-1300*. Éditions Paradigme.

David Wrisley. 2018. The Open Medieval French Initiative (OpenMedFr).

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium, October. Association for Computational Linguistics.