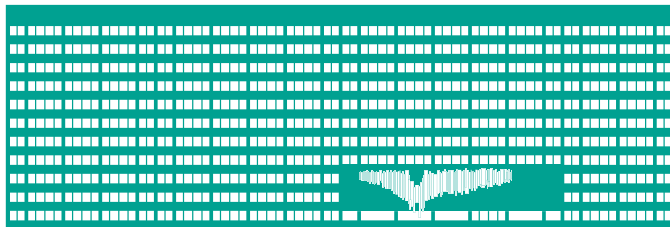


VŠB TECHNICKÁ
UNIVERZITA
OSTRAVA

VSB TECHNICAL
UNIVERSITY
OF OSTRAVA



www.vsb.cz

Kompresa stromových struktur

Semestrální projekt

Marek Beran

VŠB – Technická univerzita Ostrava

marek.beran.st@vsb.cz

23. května 2025



- 1 Úvod
- 2 Implementace knihovny
- 3 Převod textu do stromové struktury
- 4 Algoritmy
- 5 Literatura



- Je možné efektivně komprimovat textové data převedením do stromové struktury?
- Cíl: Proof of Concept



Programovací jazyk a platforma:

- C# 9.0
- .NET 5.0 a vyšší
- Visual Studio 2022

Knihovny:

- UDPipe (rozpoznávání syntaktických stromů)
- MorphoDiTa (morfologická analýza)

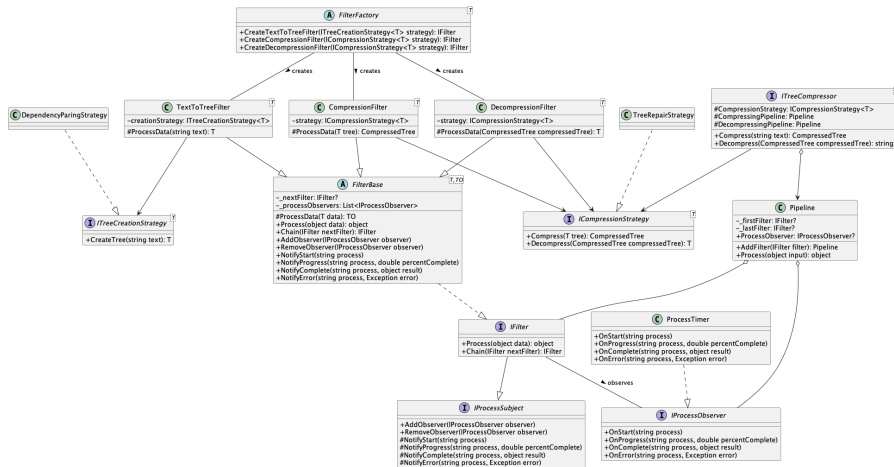
Další nástroje:

- R (datová analýza a vizualizace)
- Mkdocs (dokumentace)
- Bash skripty (podpůrné nástroje)

Bindings:

- C# wrapper pro UDPipe (nativní knihovna)
- C# wrapper pro MorphoDiTa (nativní knihovna)

Implementace knihovny

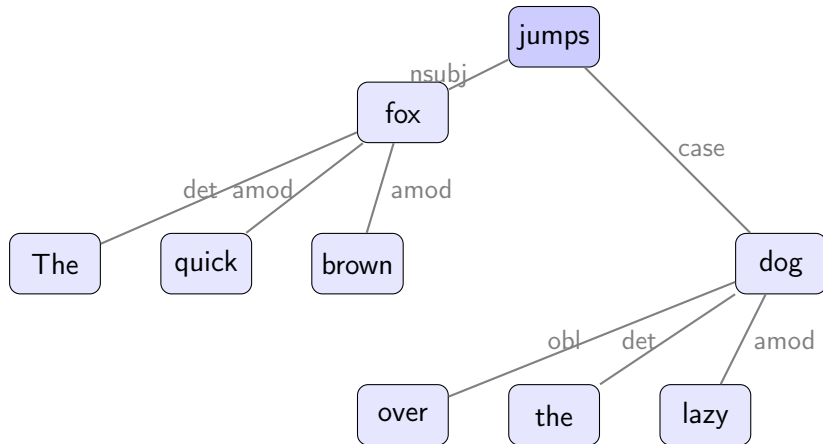


Obrázek: Třídní diagram části implementace zaměřený na řetězení filtrů

Převod textu do stromové struktury



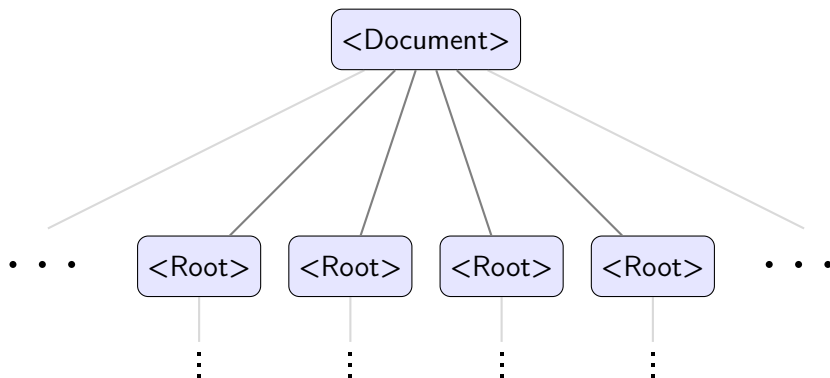
- Dependency parsing - závislosti mezi slovy ve větě.



Umělé rozšíření stromu



- Rozšíření stromu pro podporu více vět bez nutnosti práce s lesem
- Pro zajištění dostatečné velikosti syntaktického stromu pro testování a kompresi





- Gramatickou komresi, protože je vhodná pro kompresi stromových struktur, kde se často opakují podobné podstromy a vzory (např. v XML dokumentech)
- TreeRePair – algoritmus pro kompresi stromových struktur, který využívá gramatiky a pravidel pro nahrazování podstromů - využívá se pro kompresi XML a dalších stromových datových struktur
- Komprimace pomocí linearizace – převod stromu na posloupnost uzlů a jejich následná komprese pomocí algoritmů pro kompresi textu



- [1] Katja Filippova a Michael Strube. “Dependency tree based sentence compression”. In: *Proceedings of the Fifth International Natural Language Generation Conference*. 2008, s. 25–32.
- [2] Daniel Jurafsky a James H Martin. “Speech and Language Processing: An introduction to Natural Language Processing”. In: *Computational Linguistics, and Speech Recognition with Language Models. Third Edition draft* (2024).
- [3] Sandra Kübler, Ryan McDonald a Joakim Nivre. “Dependency parsing”. In: *Dependency parsing*. Springer, 2009, s. 11–20.
- [4] Markus Lohrey. “Grammar-based tree compression”. In: *Developments in Language Theory: 19th International Conference, DLT 2015, Liverpool, UK, July 27-30, 2015, Proceedings*. 19. Springer. 2015, s. 46–57.



- [5] Markus Lohrey, Sebastian Maneth a Roy Mennicke. “Tree structure compression with repair”. In: *2011 Data Compression Conference*. IEEE. 2011, s. 353–362.
- [6] Markus Lohrey, Sebastian Maneth a Roy Mennicke. “XML tree structure compression using RePair”. In: *Information Systems 38.8* (2013), s. 1150–1167.
- [7] Colt McAnlis a Aleks Haecky. *Understanding compression. data compression for modern developers*. 1st Edition. Sebastopol, CA: O'Reilly, 2016. ISBN: 978-1-491-96153-7.
- [8] F. Oquendo, J. Leite a T. Batista. “Pipe-filter architectural style”. In: *Undergraduate Topics in Computer Science* (2016), s. 171–177. DOI: 10.1007/978-3-319-44339-3_13.



- [9] Milan Straka a Jana Straková. “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*. Vancouver, Canada: Association for Computational Linguistics, srp. 2017, s. 88–99. URL: <http://www.aclweb.org/anthology/K/K17/K17-3009.pdf>.
- [10] Jana Straková, Milan Straka a Jan Hajič. “Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition”. In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, Maryland: Association for Computational Linguistics, čvn. 2014, s. 13–18. URL: <http://www.aclweb.org/anthology/P/P14/P14-5003.pdf>.

Děkuji za pozornost

Marek Beran

VŠB – Technická univerzita Ostrava

marek.beran.st@vsb.cz

23. května 2025