CSE 6324

Advanced Topics in Software Engineering

Semantic Code Search

Iteration 2

Team 7

Fotios Lygerakis

Mohammad Rifat Arefin

GitHub Repository:
https://github.com/rifatarefin/semantic-code-search

Project Plan

Features: Iteration 1

- Train a 1D-CNN model
  - Embed code and query in a joint vector space
  - Retrieve Code with the most similar vector with the vector representation of query
  - Only on Python data of the CodeSearchNet [4] dataset
- Build a command-line code search tool
  - A Jupyter notebook to perform demo code search

Features: Iteration 2

- Use the state of the art ML model from iteration 1
  - Neural bag of words model
- Train on all available six programming languages
  - Python, Javascript, Ruby, Go, Java, and PHP
- Command line code search tool for demo purpose
  - Supports programming language selection with query

Features: Planned for Future Iterations (Iteration 3)

1. Exploit State of the art language models for code search
    - Generative Pretrained Transformer(GPT) 2&3
        - Unsupervised technique
        - Transformer-based
        - Not used for semantic code search yet
        - GPT-3 Model not published yet
    - Code2Vec [11]
        - Promising for capturing code semantics

Features: Planned for Future Iterations (Iteration 3)

---

2. Develop a web app and host it online

Features: Planned for Future Iterations (Final Iteration)

---

3. Comparison between the two approaches:
    - Code & Query Encoders developed <u>separately</u> [3]
        - Code Encoder -> Representation mapped to the vector space of the Natural language model
    - End-2-End training for Code & Query Encoders [4]
        - Loss function is being calculated jointly.
4. Least priority:
    - Train a language model with Stackoverflow [7] data

Competitors

- Information retrieval based approach:
    - Reformulate queries with natural language phrasal representations of method signatures [14]
    - Recommend reformulation strategy based on query properties: uses ML [15]
    - Extend a query with synonyms generated from WordNet [16]
- Considering data and evaluation metrics:
    - Leaderboard of Code Search Net Challenge [6]

Risks: Already Encountered

1. Insufficient Hardware Resources
2. Very Large size of data

Solution

Set up the project on TACC clusters: Maverick2 [12]

- Does not provide root privilege
- Migrated to Singularity [13] from Docker for containerized environment: additional 15 hours of work

Risks (Current)

3. Insufficient/Improper Data & Modelling Techniques

    - Redirect efforts towards improving the database retrieval system
    - Developing a good UI
    - Probability: 50%, Risk effect: 40 hrs, Risk exposure: 20 hrs

4. Performance Deterioration when when adding more programming languages

    - Fine-tune models
    - Probability: 40%, Risk effect: 30 hrs, Risk exposure: 12 hrs

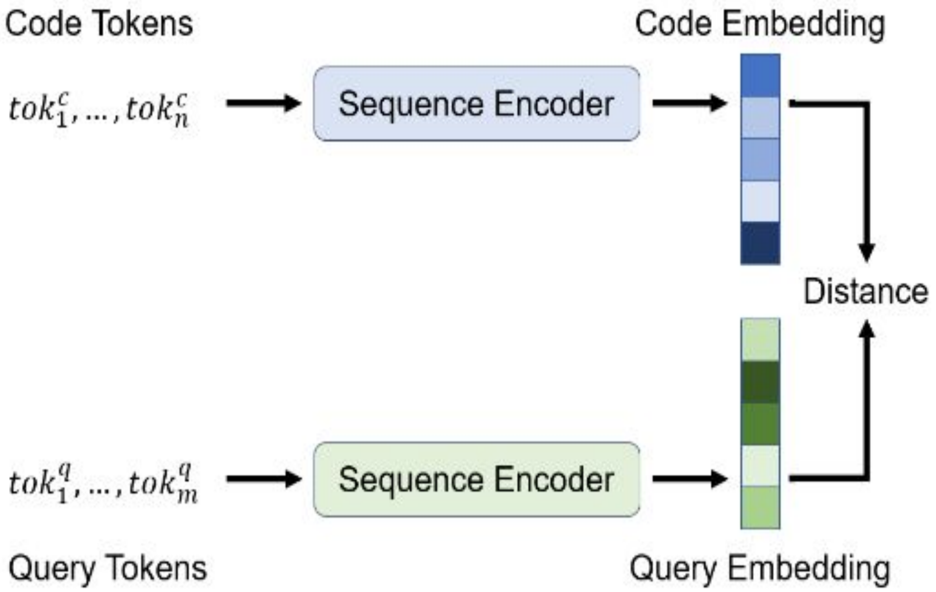
Risks (Current)

5. Host a web app version online
    - Hosting service -> must have adequate resources
    - Probability: 70%, Risk effect: 15 hrs, Risk exposure: 10.5 hrs
6. Adding additional packages in Singularity containers
    - Rebuild containers
    - Probability: 50%, Risk effect: 6 hrs, Risk exposure: 2 hrs

Specifications & Design

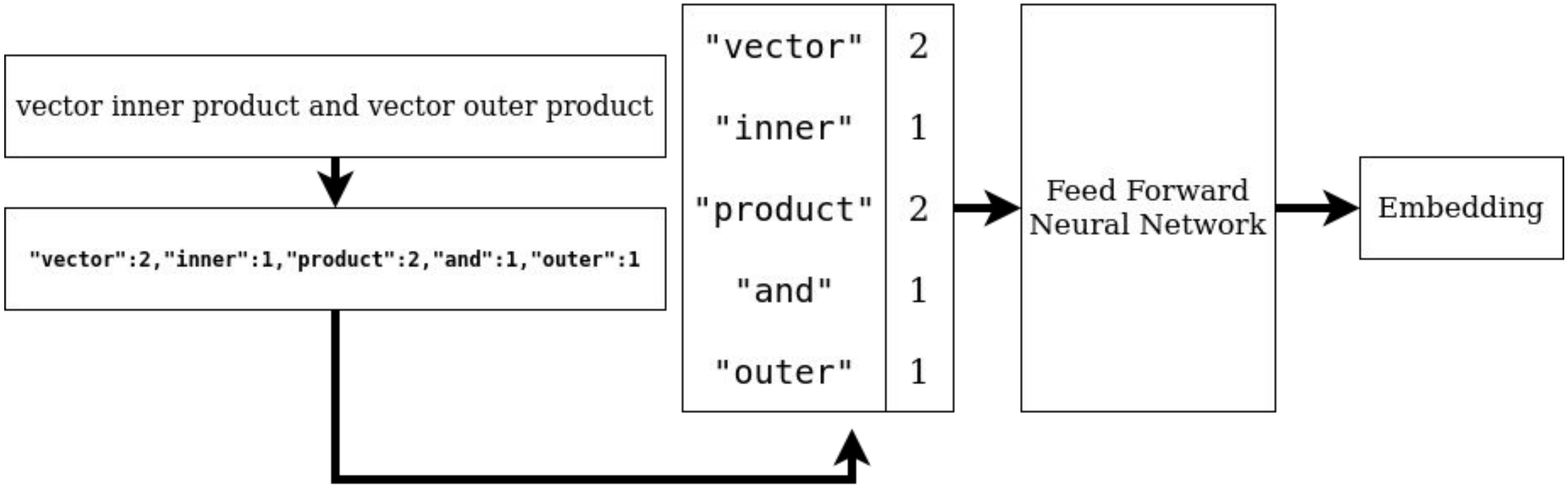
Use-Case

- Query Token Sequence
- Code Snippet Token Sequence
- Cosine Distance
- Return Code Snippet
  - Smaller Distance from the Query



[4]

Method - Neural Bag Of Words (NBOW)

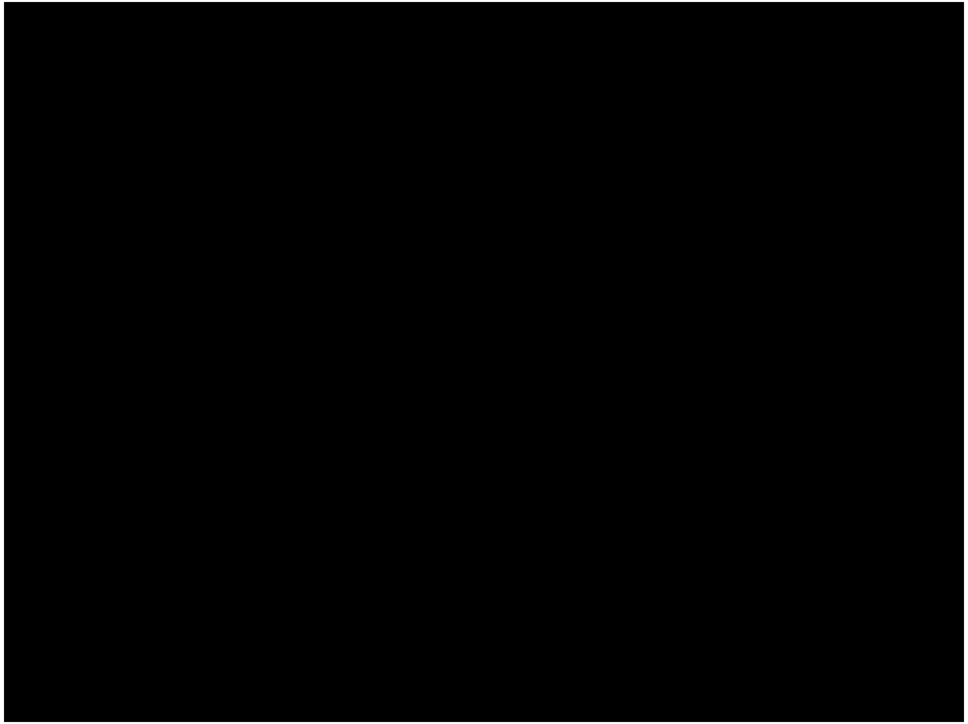


Optimization[4]:

- Max (QueryEmbedding * CodeEmbeddings)
- Min (CodeSnippetEmbedding * DistractorCodeEmbedding)

Testing

Testing - NBOW model (state-of-the-art)



Customers and Users

Customers & Users

- Code Hosting & Versioning services
  - Github, Gitlab, etc
- General Purpose Search Engines
  - Google, Bing, etc
- IDEs with integrated code search engines

Feedback

Feedback

- Initial User Experience Feedback (Iteration 1&2)
    - Team Members
- Future User Experience Feedback (Iteration 3)
    - Classmates
- Project Management Feedback
    - Project Mentor

References

References

[1] Daniel Cer and Yinfei Yang and Sheng-yi Kong and Nan Hua and Nicole Limtiaco and Rhomni St. John and Noah Constant and Mario Guajardo-Cespedes and Steve Yuan and Chris Tar and Yun-Hsuan Sung and Brian Strope and Ray Kurzweil (2018). Universal Sentence EncoderCoRR, abs/1803.11175.

[2] Stephen Merity and Nitish Shirish Keskar and Richard Socher (2017). Regularizing and Optimizing LSTM Language ModelsCoRR, abs/1708.02182.

[3] https://github.blog/2018-09-18-towards-natural-language-semantic-code-search/

[4] Hamel Husain, Ho-Hsiang Wu, Tiferet Gazit, Miltiadis Allamanis, Marc Brockschmidt: CodeSearchNet Challenge: Evaluating the State of Semantic Code Search. CoRR abs/1909.09436 (2019)

[5] Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D. & Sutskever, I. (2018), 'Language Models are Unsupervised Multitask Learners', OpenAI blog 1.8 (2019): 9.

References

[6] https://app.wandb.ai/github/codesearchnet/benchmark

[7] https://github.com/LittleYUYU/StackOverflow-Question-Code-Dataset

[8] https://github.com/hamelsmu/code_search/blob/master/notebooks/2%20-%20Train%20Function%20Summarizer%20With%20Keras%20%2B%20TF.ipynb

[9] https://github.com/hamelsmu/code_search/blob/master/notebooks/3%20-%20Train%20Language%20Model%20Using%20FastAI.ipynb

[10] https://github.com/spotify/annoy

[11] Uri Alon, Meital Zilberstein, Omer Levy, and Eran Yahav. 2019. Code2vec: learning distributed representations of code. Proc. ACM Program. Lang. 3, POPL, Article 40 (January 2019), 29 pages. DOI:https://doi.org/10.1145/3290353

[12] https://portal.tacc.utexas.edu/user-guides/maverick2

References

[13] https://sylabs.io/guides/3.0/user-guide/quick_start.html

[14] E. Hill, L. Pollock, and K. Vijay-Shanker. Improving source code search with nat-ural language phrasal representations of method signatures. InProceedings of the2011 26th IEEE/ACM International Conference on Automated Software Engineering,pages 524–527. IEEE Computer Society, 2011.

[15] S. Haiduc, G. Bavota, A. Marcus, R. Oliveto, A. De Lucia, and T. Menzies. Au-tomatic query reformulations for text retrieval in software engineering. InProceedings of the 2013 International Conference on Software Engineering, pages842–851. IEEE Press, 2013.

[16] M. Lu, X. Sun, S. Wang, D. Lo, and Y. Duan. Query expansion via wordnet foreffective code search. In2015 IEEE 22nd International Conference on SoftwareAnalysis, Evolution, and Reengineering (SANER), pages 545–549. IEEE, 2015.

Thank you!

Tech Comics: "The Software Project, Pt. 1"