DiffSearch: A Scalable and Precise Search Engine for Code Changes

Luca Di Grazia
Department of Computer Science, University of Stuttgart, Germany
luca.di-grazia@iste.uni-stuttgart.de

Paul Bredl
Department of Computer Science, University of Stuttgart, Germany
paulbredl@gmx.de

Michael Pradel
Department of Computer Science, University of Stuttgart, Germany
michael@binaervarianz.de

To benefit from the immense knowledge stored in version histories, practitioners and researchers often want to search for specific kinds of code changes. However, efficiently and effectively searching through a large amount of code changes is a non-trivial challenge.

To address this challenge, we present DiffSearch [1], a scalable and precise search engine for code changes. Our approach is based on three key ideas. First, a query language that extends the underlying programming language, making it easy to use while allowing users to express complex queries. Second, an efficient and scalable retrieval component that finds relevant code changes without linearly scanning entire version histories. Finally, an algorithm that ensures the precision of the search results w.r.t. the given query.

To better illustrate the problem and how DiffSearch addresses it, consider the following sample query. The query searches for code changes that swap the arguments passed to a call that is immediately used in a conditional. For example, such a query could be used to find fixes for swapped argument bugs.

```
      if(ID<1>(EXPR<1>, EXPR<2>)){
              <...>
→     if(ID<1>(EXPR<2>, EXPR<1>)){
              <...>
```

Our query language is an extension of the target programming language, Java in the example, and adds placeholders for some syntactic categories. For example, the `ID<1>` placeholder matches any identifier, and the `EXPR<1>` placeholder matches any expression. Instead of such placeholders, queries can also include concrete identifiers and literals, for example, to search for specific API changes.

As a set of code changes to search through, suppose we have the following three examples:

Code change 1:

```
if(check(a - 1, b)){  →  if(check(a - 1, c)){
```

Code change 2:

```
if(isTruePoint(x, y)){ → if(isTruePoint(y, x)){
```

Code change 3:

```
while(var > k - 1){   → while(var > k){
```

Only the second of the three code changes matches the query and will hence be retrieved by DiffSearch.

We design our approach for several usage scenarios. One scenario is that developers index their own repositories and search for code changes they are interested in, such as an API migration performed years ago. Another scenario is that developers search millions of code changes across various open-source projects for changes to learn from, such as common refactorings. Finally, DiffSearch can help to find examples for training a neural model or to be used in few-shot learning.

We perform a large-scale evaluation of DiffSearch on one million code changes for each of three currently supported programming languages: Java, Python, and JavaScript. We find that DiffSearch retrieves relevant code changes with a recall of 80.7% for Java, 89.6% for Python, and 90.4% for JavaScript. Moreover, we perform a user study to compare DiffSearch with a regular expression-based search and with GitHub Search, which shows users to be more productive when using DiffSearch. Finally, we illustrate in a case study that the approach can retrieve 74,903 bug fixes for common fix patterns defined in previous work.

In summary, our work paper contributes the following:

- A *query language* that extends the target programming language with placeholders and wildcards, making it easy to adapt the approach to different languages.
- A technique for searching for code changes that ensures *scalability* through approximate, indexing-based retrieval, and that ensures *precision* via exact matching.
- Empirical evidence that the approach effectively finds thousands of relevant code changes, scales well to more than a million changes from different projects, and successfully helps users answer a diverse set of queries.

Useful:
Original Paper: 10.1109/TSE.2022.3218859
Official website: http://diffsearch.software-lab.org
Presenter: Luca Di Grazia

References

[1] L. D. Grazia, P. Bredl, and M. Pradel, "Diffsearch: A scalable and precise search engine for code changes," *IEEE Transactions on Software Engineering*, vol. 49, no. 4, pp. 2366–2380, 2023.