Learning And Mining from DatA
http://lamda.nju.edu.cn

Automatic Code Review by Learning the Revision of Source Code

Shu-Ting Shi[1], Ming Li[1,2], David Lo[3], Ferdian Thung[3], Xuan Huo[1]

[1]National Key Laboratory for Novel Software Technology, Nanjing University

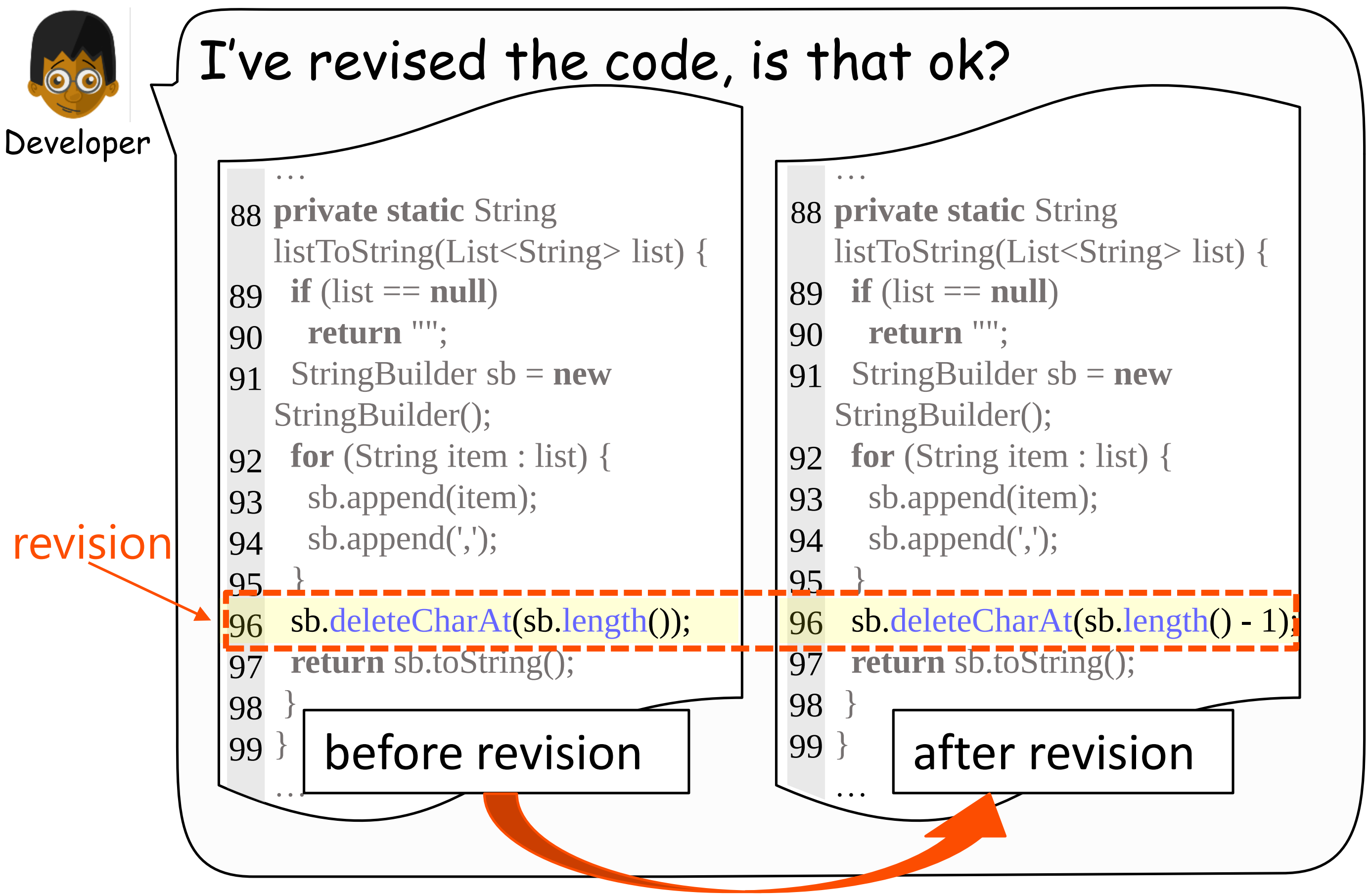
[2]Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University

[3]School of Information Systems, Singapore Management University, Singapore

{shist, lim}@lamda.nju.edu.cn

Code keeps on changing

LAMDA
Learning And Mining from DatA
http://lamda.nju.edu.cn



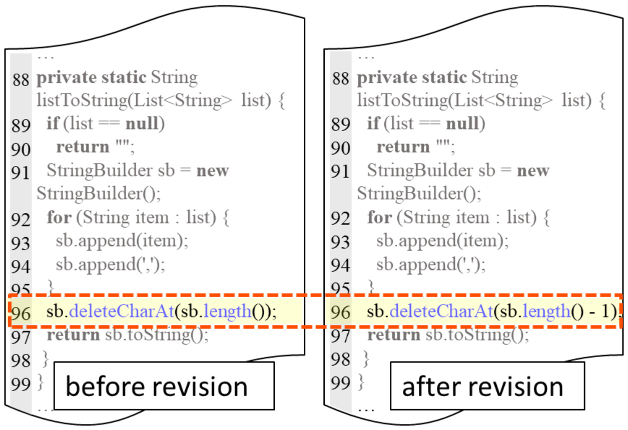
How to determine?

Key point:
Correctly **model the revision** between source code.

Modeling the revision is hard

The state-of-the-art software mining approaches leverage the ***global*** information by modeling the relationship between two pieces of code.

But apparently, model the revision needs to emphasis the ***local*** information, i.e., the revised small portion of codes.



Key challenge: model the revision of source code and avoid being misled by the overwhelming among of unchanged code in the revision.

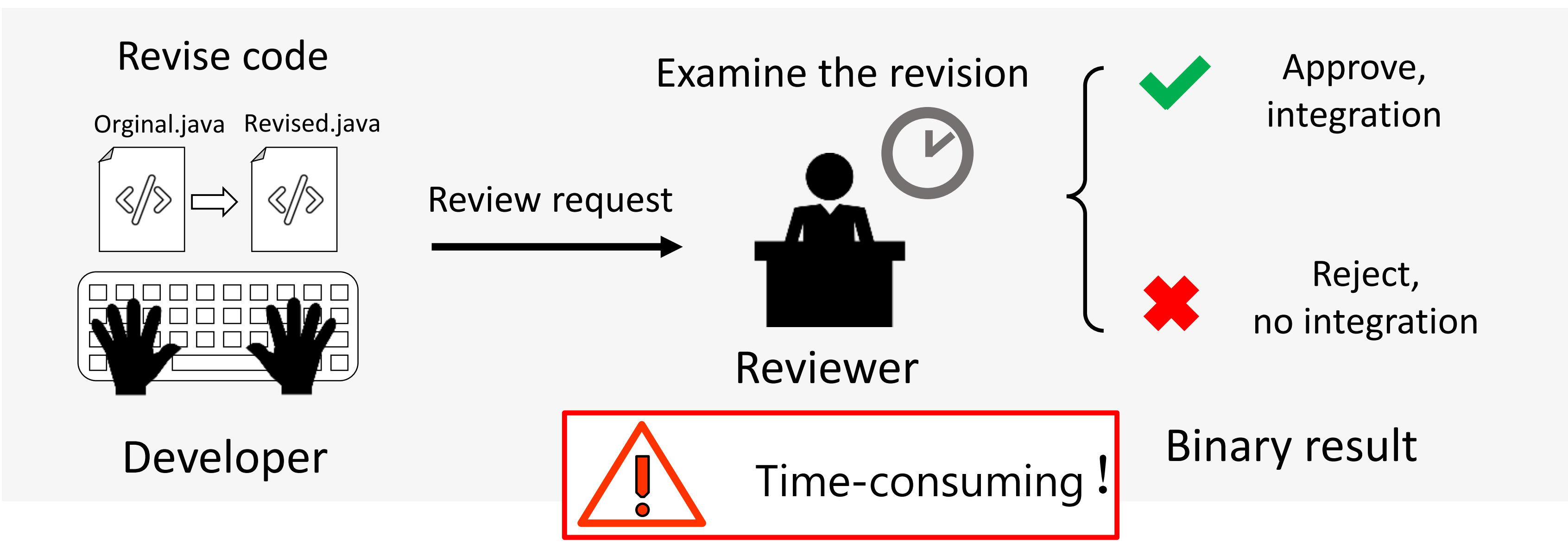
This work

Outline

- What is code review?
- The proposed approach DACE
- Experiments
- Conclusion

Code Review

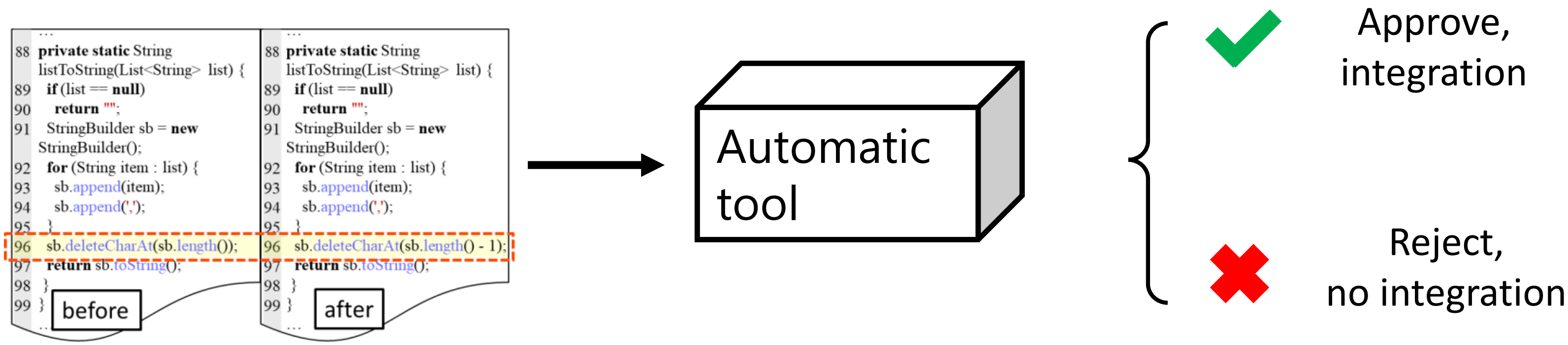
Code review is the process of inspection on two versions of a source code in order to improve the overall software quality and ensure the security of the project.



automating such the code review process will alleviate the burden of code reviewers and speed up the software maintenance process.

Automatic Code Review

http://lamda.nju.edu.cn



Key challenge: how to model the revision.

Exiting models for modeling the relationship between software artifacts

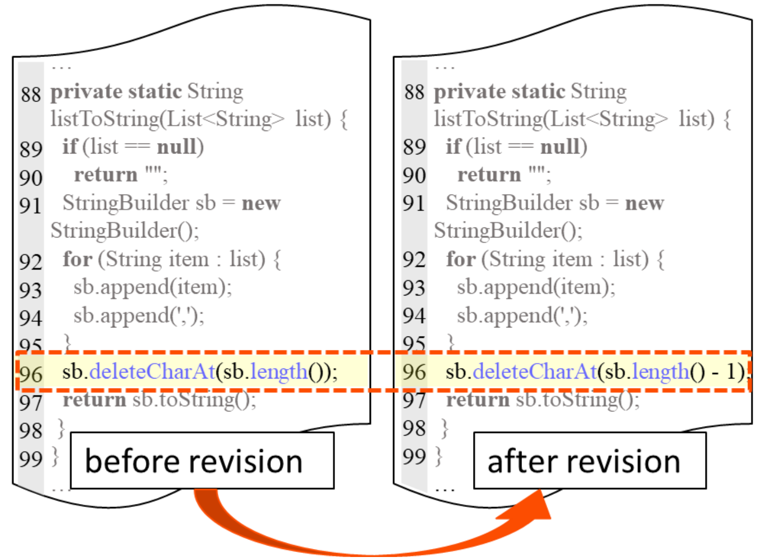
- **NPCNN** [Huo and Li, 2017]
- **CDLH** [Wei and Li, 2017]
- **LSCNN** [Huo and Li, 2018]
- **CDPU** [Wei and Li, 2018]

*these approaches are equipped for extract **global** feature to model the "**similarity**".*

But none of them can focus on the **local** feature to model the **"difference"**.

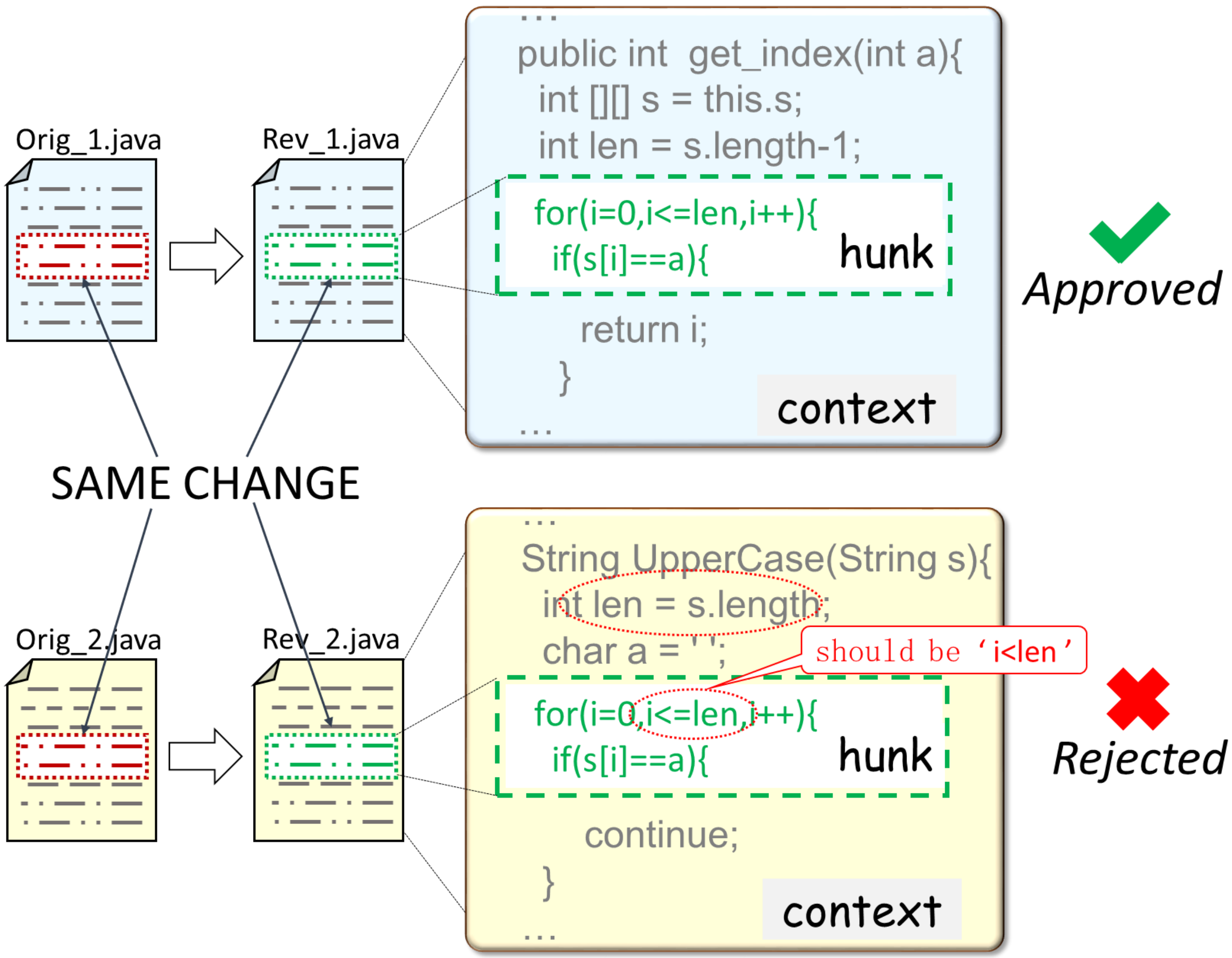
Challenge 1

How to model the revision of source code?



Key idea: A particularly designed pairwise autoencoder (PAE) is employed to learn the revision features. We construct an encoder to pact the revised code into a compact feature representations based on which the transformation can be learned.

Challenge 2



Same change different result.

Context of the ***changed hunks*** provides rich information about the revision.

Challenge 2

LAMDA
Learning And Mining from DatA
http://lamda.nju.edu.cn

- The changed hunk is usually ***short,*** while the context is usually ***long***.
- **Dilemma**
  - Utilize all the context？
    overwhelmed by same unchanged code lines. ✖
  - Discard the context？
    the revised code is too short to provide sufficient information for modeling . ✖

How to **focus on modeling the revision but also considering the rich information in the context?**

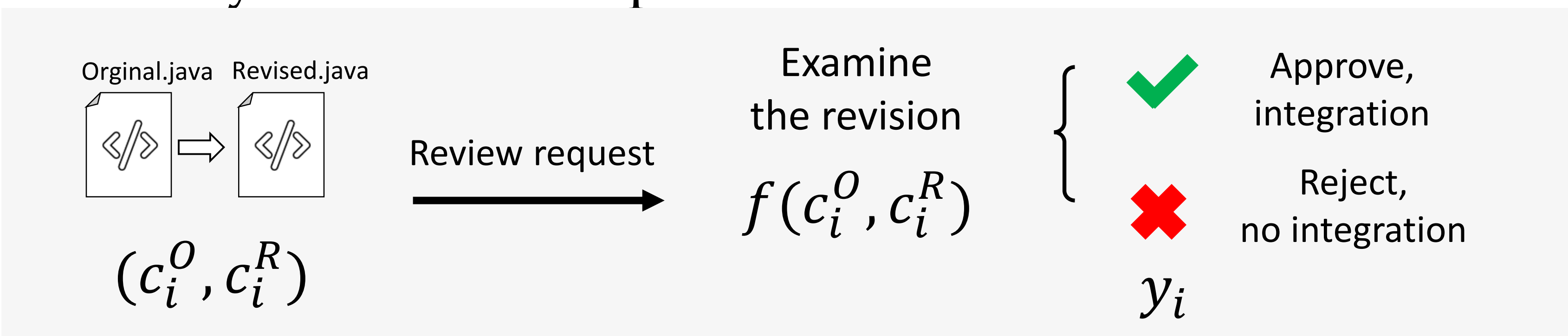
Key idea: Enrich the revised lines with the information of context around them.

Outline

- What is code review?
- The proposed approach DACE
- Experiments
- Conclusion

Formalization

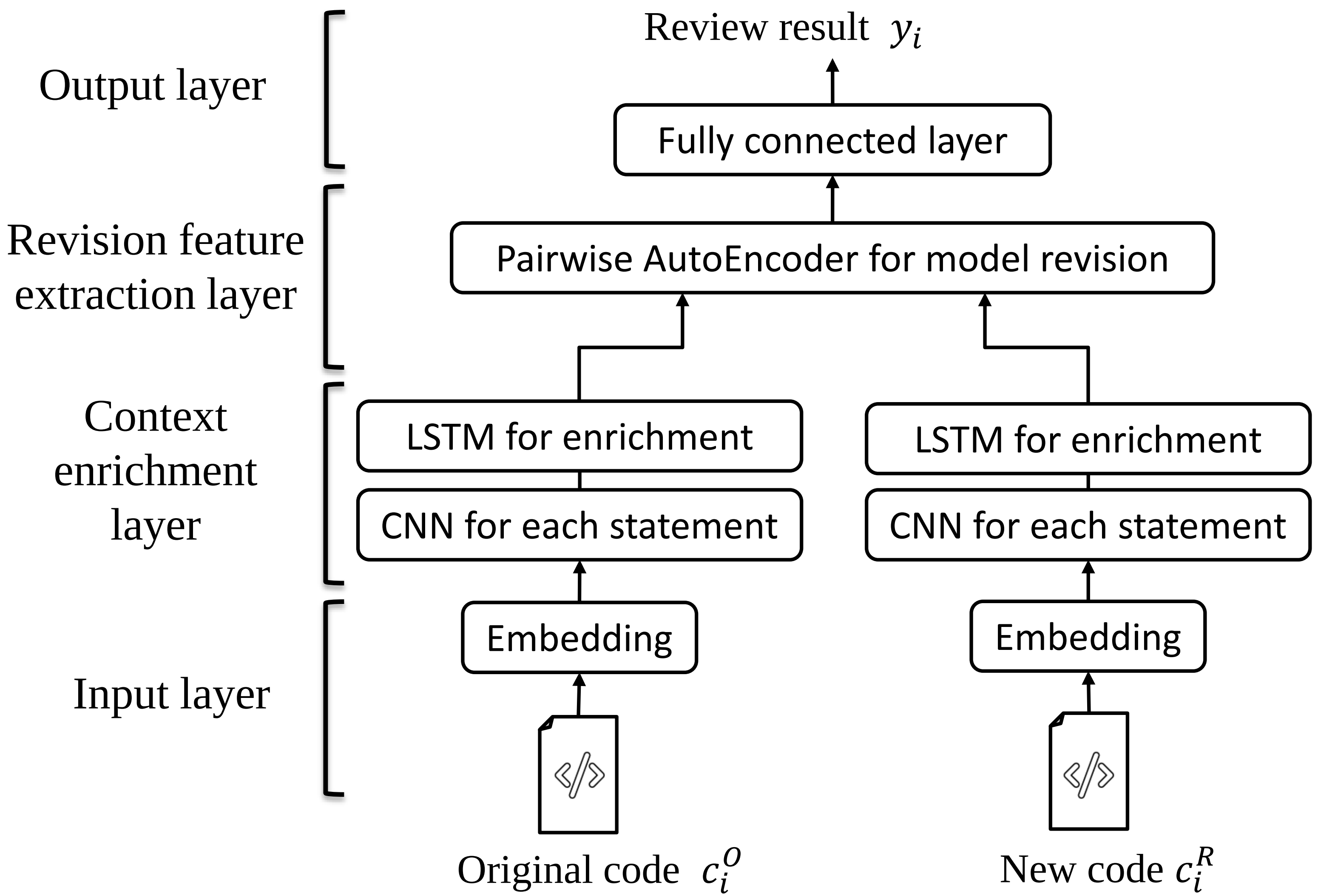
- We formalize the code review as a learning task, which is a binary classification problem.



  - Given a sample of data: (c_i^O, c_i^R, y_i), $c_i^O \in \mathcal{C}^O, c_i^R \in \mathcal{C}^R$
  - Learn a mapping $f: \mathcal{C}^O \times \mathcal{C}^R \longmapsto \mathcal{Y}$ that can be learned by minimizing the objective function:

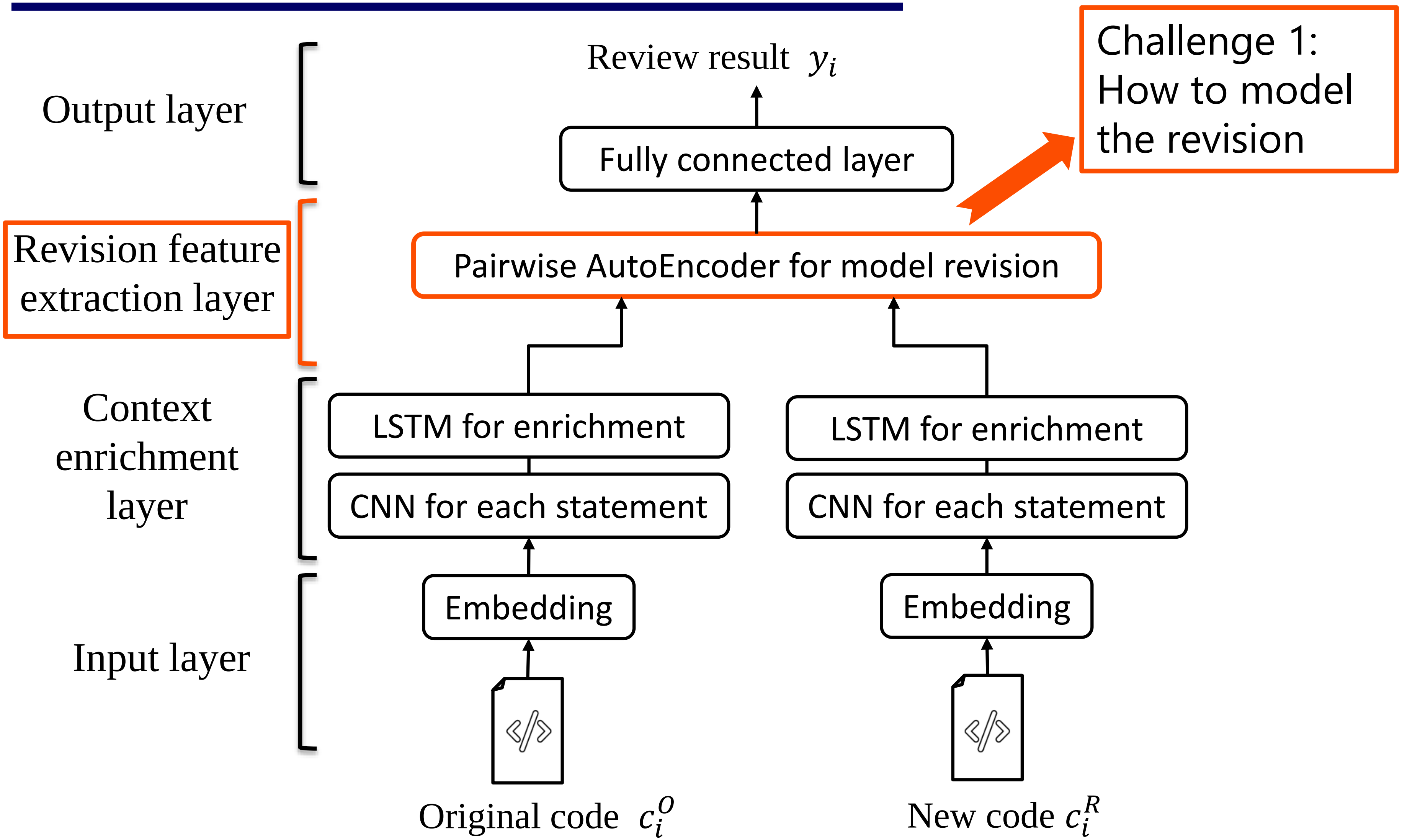
$$\min_f \sum_i \mathcal{L}\left(f\left(c_i^O, c_i^R\right), y_i\right) + \lambda \Omega(f)$$

Model Structure of DACE



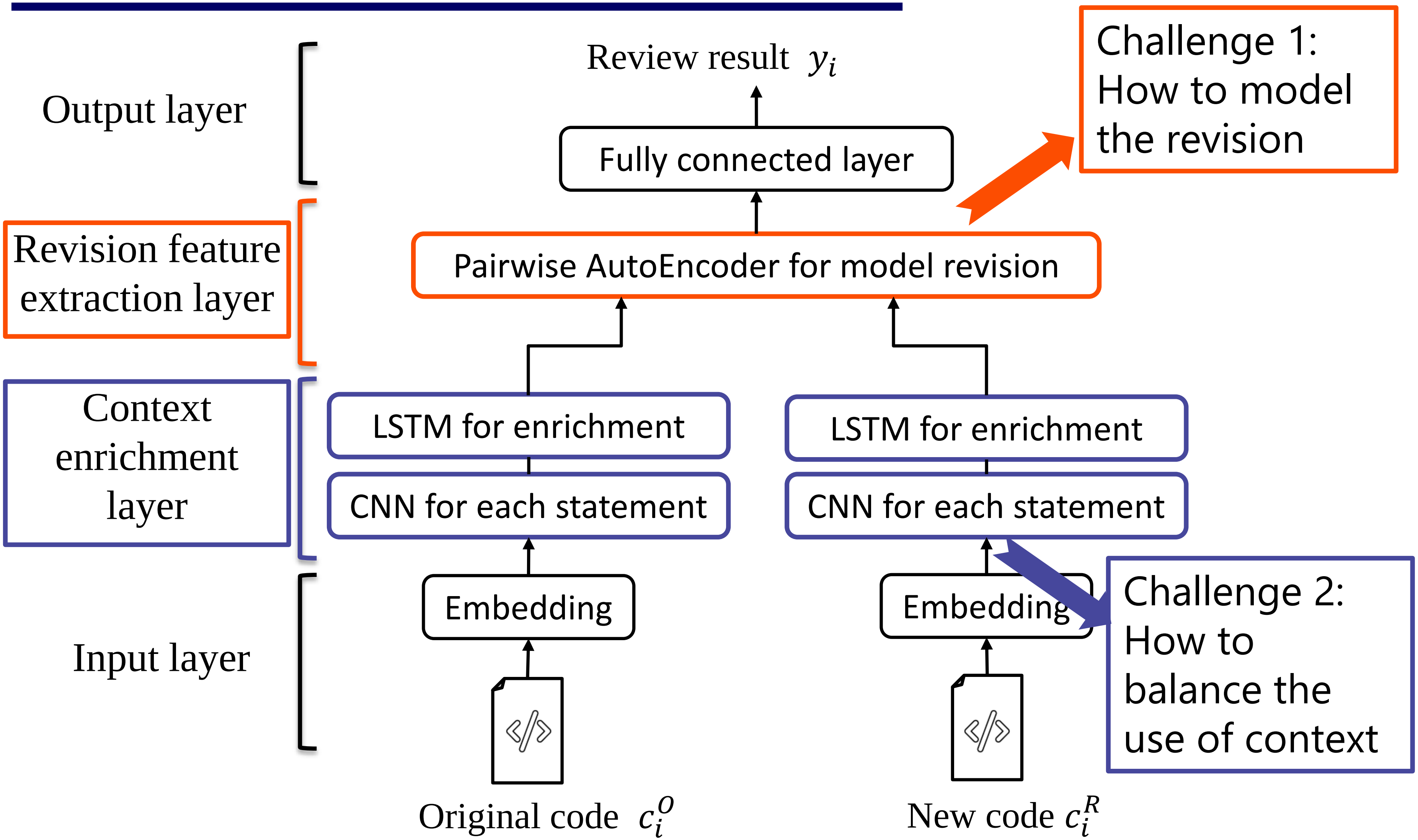
Model Structure

LAMDA Learning And Mining from DatA
http://lamda.nju.edu.cn



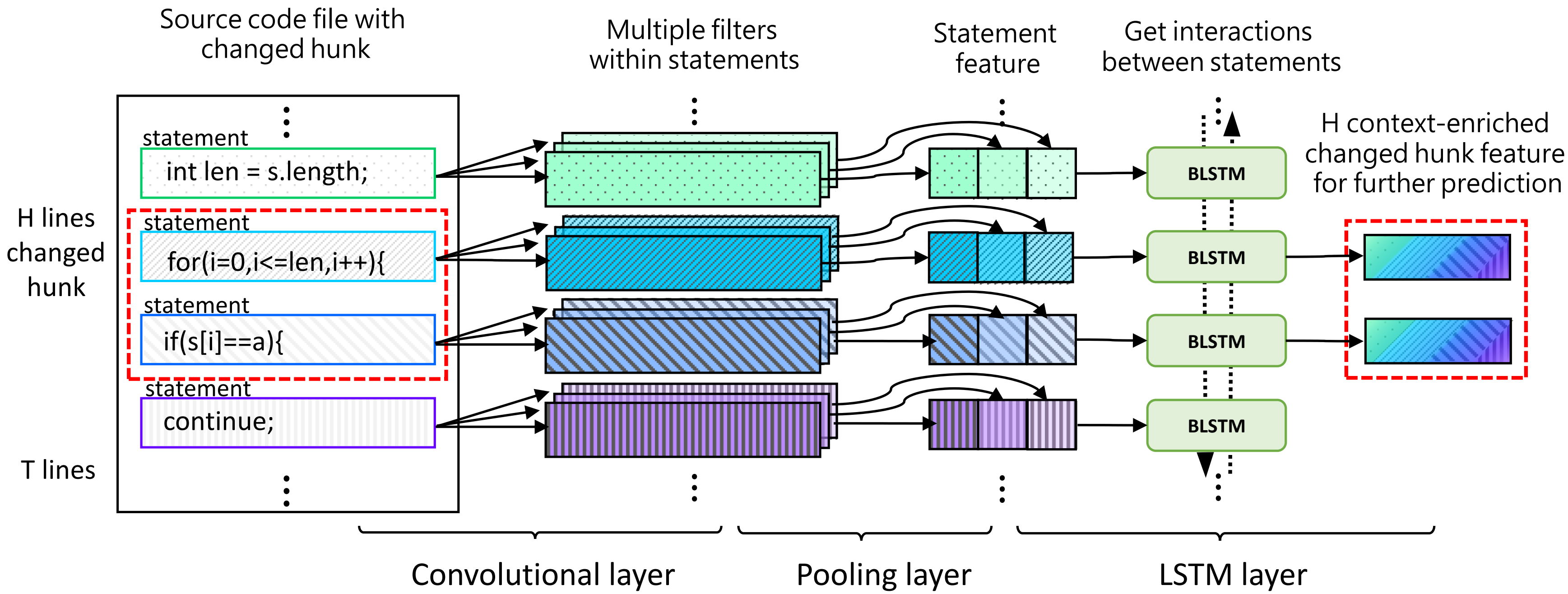
Model Structure

Learning And Mining from DatA
http://lamda.nju.edu.cn

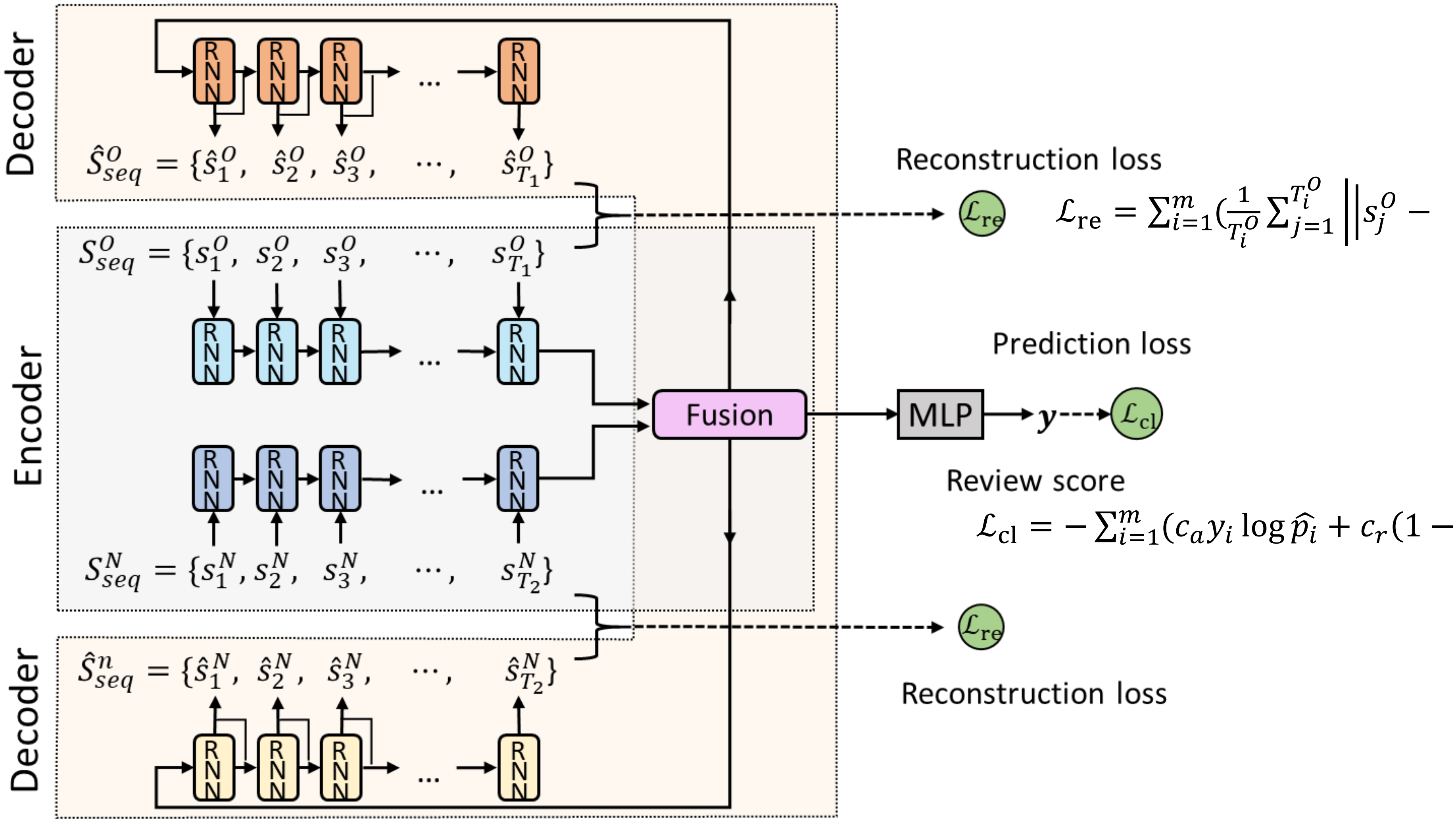


Context Enrichment Layer

- Extract the statement level features by convolutional layer
- Enrich the current statement with other statements in the context by exploiting the sequential relationship by LSTM layer



Revision Feature Extraction Layer



Outline

Learning And Mining from DatA
http://lamda.nju.edu.cn

- What is code review?
- The proposed approach DACE
- Experiments
- Conclusion

Experiment Setting

- Dataset
  - The code review data used in the experiments is crawled from Apache Code Review Board。
  - For each changed hunk, if it has the highlighted lines that marked by reviewers denoting they have issues, we regard the hunk as rejected.

Repository	#hunk
accumulo	5,620
ambari	6,810
aurora	6,762
cloudstack	6,171
drill-git	3,575
hbase-git	6,702

- Evaluation measure:
  - ✓ AUC
  - ✓ F1

Experiment

- TFIDF-LR : uses TFIDF to represent the original and the revised source code, and LR is used for prediction. [Schütze *et al.*, 2008][Gay *et al.*, 2009]
- TFIDF-SVM: like above but SVM is used for prediction. [Schütze *et al.*, 2008][Gay *et al.*, 2009]

Non-deep methods

- Deeper : a state-of-the-art model on software engineering, which uses basic features for changes. [Yang *et al.*, 2015]
- Deeper-SVM: like above but SVM is used for prediction. [Yang *et al.*, 2015]

Deep methods

- LSCNN: a variant of DACE that without PAE. It is similar to the state-of-art method LSCNN and using Multilayer Perceptron (MLP) for prediction.
- PAE : a variant of DACE that only use PAE to extract features from CNN without considering sequential and long-term dependency information between statements.

Variants

Experiment

LAMDA
Learning And Mining from DatA
http://lamda.nju.edu.cn

Repository	TFIDF-LR		TFIDF-SVM		Deeper		Deeper-SVM		LSCNN		PAE		DACE	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC
accumulo	0.227	0.666	0.239	0.703	0.202	0.688	0.199	0.705	0.417	0.787	0.373	0.814	0.493	0.786
ambari	0.240	0.708	0.278	0.848	0.306	0.680	0.238	0.572	0.444	0.824	0.473	0.861	0.509	0.905
aurora	0.204	0.582	0.220	0.645	0.349	0.682	0.299	0.564	0.336	0.750	0.571	0.819	0.403	0.793
cloudstack	0.250	0.745	0.275	0.827	0.352	0.795	0.265	0.646	0.360	0.761	0.415	0.820	0.516	0.852
drill-git	0.212	0.658	0.236	0.725	0.229	0.593	0.212	0.540	0.318	0.788	0.382	0.806	0.573	0.820
hbase-git	0.232	0.679	0.256	0.759	0.193	0.590	0.154	0.524	0.348	0.751	0.411	0.764	0.396	0.813
average	0.228	0.673	0.251	0.751	0.272	0.671	0.228	0.592	0.370	0.777	0.438	0.814	0.482	0.828

DACE outperforms other state-of-the-art competing methods in term of F1 and AUC.

Experiment

Learning And Mining from DatA
http://lamda.nju.edu.cn

Repository	TFIDF-LR		TFIDF-SVM		Deeper		Deeper-SVM		LSCNN		PAE		DACE	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC
accumulo	0.227	0.666	0.239	0.703	0.202	0.688	0.199	0.705	0.417	0.787	0.373	0.814	0.493	0.786
ambari	0.240	0.708	0.278	0.848	0.306	0.680	0.238	0.572	0.444	0.824	0.473	0.861	0.509	0.905
aurora	0.204	0.582	0.220	0.645	0.349	0.682	0.299	0.564	0.336	0.750	0.571	0.819	0.403	0.793
cloudstack	0.250	0.745	0.275	0.827	0.352	0.795	0.265	0.646	0.360	0.761	0.415	0.820	0.516	0.852
drill-git	0.212	0.658	0.236	0.725	0.229	0.593	0.212	0.540	0.318	0.788	0.382	0.806	0.573	0.820
hbase-git	0.232	0.679	0.256	0.759	0.193	0.590	0.154	0.524	0.348	0.751	0.411	0.764	0.396	0.813
average	0.228	0.673	0.251	0.751	0.272	0.671	0.228	0.592	0.370	0.777	0.438	0.814	0.482	0.828

The internal structure of pairwise autoencoder helps DACE to captures the revision by learning the revision patten by PAE.

Experiment

Learning And Mining from DatA
http://lamda.nju.edu.cn

Repository	TFIDF-LR		TFIDF-SVM		Deeper		Deeper-SVM		LSCNN		PAE		DACE	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC
accumulo	0.227	0.666	0.239	0.703	0.202	0.688	0.199	0.705	0.417	0.787	0.373	0.814	0.493	0.786
ambari	0.240	0.708	0.278	0.848	0.306	0.680	0.238	0.572	0.444	0.824	0.473	0.861	0.509	0.905
aurora	0.204	0.582	0.220	0.645	0.349	0.682	0.299	0.564	0.336	0.750	0.571	0.819	0.403	0.793
cloudstack	0.250	0.745	0.275	0.827	0.352	0.795	0.265	0.646	0.360	0.761	0.415	0.820	0.516	0.852
drill-git	0.212	0.658	0.236	0.725	0.229	0.593	0.212	0.540	0.318	0.788	0.382	0.806	0.573	0.820
hbase-git	0.232	0.679	0.256	0.759	0.193	0.590	0.154	0.524	0.348	0.751	0.411	0.764	0.396	0.813
average	0.228	0.673	0.251	0.751	0.272	0.671	0.228	0.592	0.370	0.777	0.438	0.814	0.482	0.828

PAE only not perform better than DACE because it abandons the context of revised hunk which is indispensable.

Outline

- What is code review?
- The proposed approach DACE
- Experiments
- Conclusion

Conclusion

- Major Contribution
  - Put forward a new software mining problem that aims to model the relationship between two source files by **learning the revision features.**
  - Propose a novel model DACE, which learns the revision features by modeling the hidden state, based on pairs of context-enriched representation of source code.

- Future work
  - Considering the relationship between hunks.
  - Incorporating additional data.
  - …

Thanks!