3 Related work [≈ 0.5 pages]

You should find existing papers, group them into categories based on their approaches, and discuss their strengths and weaknesses, as well as how they are similar to and differ from your work. In your opinion, which approaches were clever/good? What is the state- of-the-art? Do most people perform the task by hand? You should aim to have at least 5 references in the related work. Include previous attempts by others at your problem, previous technical methods, or previous learning algorithms. Google Scholar is very useful for this: https://scholar.google.com/ (you can click “cite” and it generates MLA, APA, BibTeX, etc.)

Omondiagebe et al. (2019): **Features that Predict the Acceptability of Java and JavaScript Answers on Stack Overflow**

Model: Random forest, Neural network (5 hidden layers)

Sampling algorithm: SMOTE, ADASYN

Feature:

* Code
  + Number of code line
  + Code length: average number of identifiers (e.g., constant, parameters) per line
* Textual Features
  + Question and answer text similarity: cosine similarity
  + Question and answer code similarity: cosine similarity
  + Polarity: emotional content
  + Vector Concordance Similarity
  + Body length: number of words, ~~number of sentences~~
* Non-textual feature
  + Response time: time lag between question and answer
  + Number of comments
  + Answer Count: for question
  + Answer Score: voting
  + View Count: for answer
* Use features
  + Reputation
  + ~~Use time: sign up date~~

Performance

* Accuracy/ Precision/ Recall

|  |  |  |
| --- | --- | --- |
|  | SMOTE | ADASYN |
| Random Forest | 71.7% / 88.3% / 73.3% | 70.6% / 85.0% / 71.1% |
| Neural network | 70.9% / 87.3% / 72.2% | 69.8% / 83.1% / 69.5% |

Shao and Yan (2017): **Recommending Answerers for Stack Overflow with LDA Model**

Model: logistic regression

Gantayat et al. (2015): **The Synergy Between Voting and Acceptance of Answers on StackOverflow, or the Lack thereof**

Model: logistic regression

Dataset: 1 – accepted, 0 – not accepted but have higher votes than accepted

Feature:

* Body length (+): word count
* Code length (+): word count
* Hyperlinks (-): number
* Readability (-): Flesch-Kincaid score
* Similarity (+): cosine similarity index

Performance:

* Accuracy: 56.24%

Islam et al. (2018): **RAiTA: Recommending Accepted Answer Using Textual Metadata**

Dataset: textual features and metadata

Accuracy: 89.7%

Calefato et al. (2015): **Mining Successful Answers in Stack Overflow**

Model: logistic regression

Feature:

* Presentation quality
  + Length: # characters
  + Uppercase ratio
  + URL Count
  + Code snippets: binary
* Affect
  + Polarity/sentiment of answer
  + Polarity/sentiment of comments
* Time
  + Arrival order (ranking)
  + Elapsed time: time difference for question and answer
* Reputation
  + Answer’s reputation
  + Answer’s number of badges
  + Asker’ reputation

Performance:

* Area under curve = 0.50

Nasehi et al. (2012): **A Study of Programming Q&A in StackOverflow**

Model: qualitative analysis

Features (attributes):

* Concise code
* Similar context to question
* Highlight important elements
* Hyperlinks
* Multiple code chunks: step-by-step solution
* Multiple solutions
* Inline documentation
* Solution limitation
* API limitation