Information retrieval: "Information Retrieval is widely used to obtain proper information from source code to automatically generate natural language descriptions."

a. Keyword identification approaches

i. Salton et al. \cite{Salton1975}

ii. Blei et al. \cite{Blei2003}

iii. Landauer et al. \cite{Landauer1998}

iv. Haiduc et al. \cite{Haiduc2010}

b. Eye tracking techniques found that tested developers spend more time reading method signatures than method invocations \cite{Rodeghero2014}

c. Topic modelling techniques (hierarchical PAM) \cite{Mimno2007}

d. Static analysis

i. Rastkar \cite{Rastkar2010}

ii. Dawood et al. \cite{Dawood2017}

"Stereotypes are abstractions of methods' or classes' types and roles in software systems \cite{moreno2013automatic}. JSummariser is a tool that identified stereotypes within classes and defined different text templates for summarizing different stereotypes \cite{moreno2013automatic,abid2015using}. Specifically, summary templates were created for each method stereotype \cite{abid2015using}. Micropatterns are another type of abstraction of a method's function, and a method can be mapped to several micropatterns \cite{malhotra2018class,rai2017method}. Researchers have used various techniques to identify micropatterns, such as analyzing code dependencies \cite{malhotra2018class} or converting code to XML and then identifying micropatterns from there \cite{rai2017method}."

External descriptions can be used to generate comments for code segments. For instance, Wong et al. crawled code segments together with their descriptions from Stackoverflow to automatically generate comments \cite{wong2013autocomment}. Other researchers have analyzed existing code repositories and identified similar code within external repositories to generate comments, some of which used natural language processing techniques \cite{wong2015clocom}. Another approach is gamified crowdsourcing of code-description mappings, as proposed by Badihi and Heydarnoori in their Crowdsummarizer tool \cite{badihi2017crowdsummarizer}. Finally, Huang et al. proposed a method that mines commit-comment pairs from version control systems to generate comments \cite{huang2017mining}

Natural language processing

a. Process code using a SoftwareWord Usage Model (SWUM), to capture relationships between words within code \cite{xia2017measuring}.

b. Use of the PageRank algorithm \cite{langville2011google} to retrieve the most important methods \cite{mcburney2014automatic}.

c. Abstract Syntax Tree: "Wang et al. \cite{wang2017automatically} utilized an abstract syntax tree (AST) and operations performed on related objects to identify object-related action units in the method."

Machine learning and artificial neural networks have been applied to software engineering to automate tasks such as source code summarization \cite{fowkes2017autofolding, iyer2016summarizing, nazar2016source, rastkar2013code, zheng2017code, hu2018deep, liang2018automatic, allamanis2016convolutional}. These methods can be divided into two main categories: supervised and unsupervised learning. In supervised learning, algorithms such as Support Vector Machines (SVM) and Naïve Bayes have been used for tasks such as ranking descriptive sentences \cite{rastkar2013code} and source code fragment summarization \cite{nazar2016source}. In unsupervised learning, algorithms such as Tree-based Autofolding Software Summarisation Algorithm (TASSAL) \cite{fowkes2017autofolding}, CODE-NN \cite{iyer2016summarizing}, DeppCom \cite{hu2018deep}, Code Attention \cite{zheng2017code}, and Convolutional Attention Network \cite{allamanis2016convolutional} have been used. These models use various techniques such as recurrent neural networks (RNNs), long short-term memory (LSTM), and attention mechanisms to generate natural language comments that summarize source code. Some models outperform others, such as DeppCom which outperforms CODE-NN \cite{hu2018deep}. The Convolutional Attention Network is used to generate extremely concise summarization of code \cite{allamanis2016convolutional}.