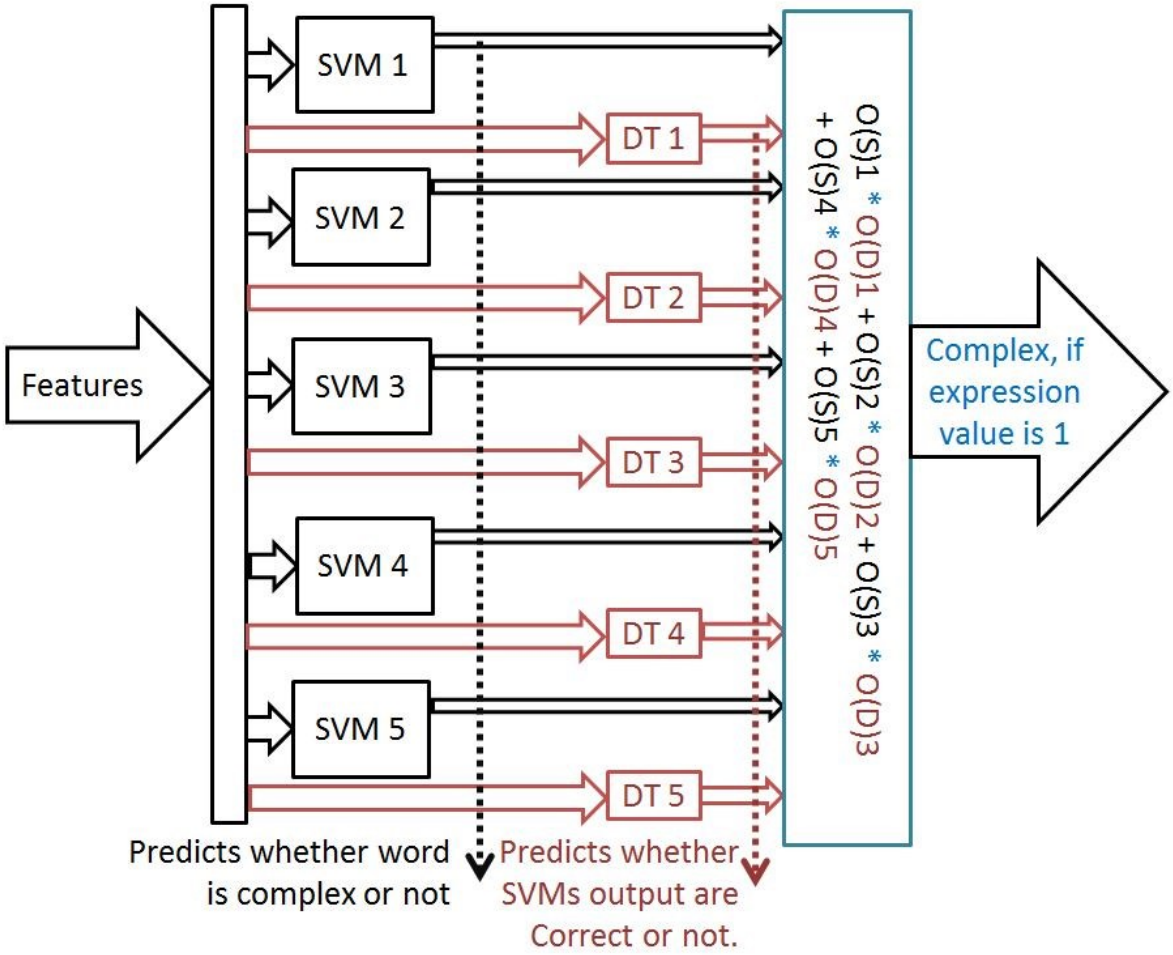
Title: System Description - CWI SemEval 2016

System 1 Name: HSVM&DT

Description

HSVM&DT is a hybrid model comprising of multiple SVMs and Decision Tree for classifying words. This system is using 5 SVMs with different training parameters (gamma, kernel, and C values) and 5 Decision Tree Classifier. SVMs perform classification of words as complex or non-complex, while DTs classify the result from SVMs as correct or incorrect. Figure below explains the architecture.



The feature vectors used for training SVMs and DTs are same as used in SVMPP model.
Training data 'cwi_training.txt' has been randomly divided into two sets in ratio of 60:40. The larger set is used to train the SVMs and smaller set is used for training DTs. Output values for DT are calculated by comparing SVMs results with actual result on smaller data set (40%). There is a decision block that operates on the output of these 10 classifiers with a function given by expression in figure and outputs the final decision on word's complexity. When evaluated on training set (40% data for SVM, 35% data for DT and 25% data for testing), this system has recall of 0.78 compared to 0.61 with one Decision Tree (data divided as 75% training and 25% testing).

Team Name: GARUDA
System 2 Name: SVMPP

Description

SVMPP is designed to get higher recall, even though system generates many False Positive values. There are 20 separate SVMs trained over data set in 'cwi_training_allannotations.txt'. Each SVM corresponds to one annotator. The features used for this system are:

- Frequency of word and it's pos-tag in training data
- Word length, vowels and consonant ratio, stem size, number of syllables
- Number of synsets with same postag as the word in sentence and the ratio (#word / #sum of all such synsets).
- Number of n-grams of characters (a-z) (100 most commonly occurring bi-,tri- grams for SVM.
- Word Position

The output of SVMs are combined as the weighted average of accuracy for each classifier. The combining function is given by:

$$\textbf{Combined Output = 1 iff } \sum_{i=1\ to\ 20} \lambda(i) * Output(i) \textbf{ >0.5, 0 otherwise}$$

$$\lambda\textbf{(i)= Accuracy of i}^{\textbf{th}}\textbf{ Classifier}$$

Model has been developed by training these classifiers on 75% data and calculating λ on 25% data (for Evaluation).