## Test Plan and Results

### Overall Test Plan

Our approach to testing will include two phases. First, we will test the requirements individually, using a protocol that has been deduced through analysis of ***Cohen's kappa***, which is a statistic that is used to measure inter-rater reliability for qualitative items. These tests will cover a variety of normal and abnormal conditions for acceptance. We will create benchmarks that will be used specifically for acceptance testing. These benchmarks will aid in identifying both normal and abnormal cases. Second, we will test the entire system (i.e. algorithm) in a production environment, using various data sets (i.e. sentences) to confirm the integrity of our automated program. This will verify the functionality of the system and provide output data that will be compared with the first phase of the test plan. These comparisons will make sure that the system can handle an increased volume of sentences being fed to the program.

### Test Case Descriptions

TL1.1 **LTL Acceptance Test 1**
TL1.2 This test will look for obvious signs of a classic temporal sentence.
TL1.3 This test will run the same protocol for testing for each requirement, manually
TL1.4 Inputs: The inputs for this test will be the documents given to us from Dr. Niu
TL1.5 Outputs: Yes/ No / Maybe
TL1.6 Normal
TL1.7 Whitebox
TL1.8 Functional
TL1.9 Unit Test
TL1.10 Results: All Sentences were parsed and categorized

TL2.1 **LTL Acceptance Test 2**
TL2.2 This test will look for less obvious examples of temporal logic
TL2.3 For this test we will run the same requirements as before, manually, and then look for implications that there is temporal logic
TL2.4 Inputs: Documents from Dr. Niu
TL2.5 Outputs: Yes/ No / Maybe
TL2.6 Normal
TL2.7 Whitebox
TL2.8 Functional
TL2.9 Unit Test
TL2.10 Results: All sentences are re-parsed and re-categorized

TL3.1 **LTL Acceptance Test 3**
TL3.2 This test will parse through the phrases with BOOL values in them
TL3.3 We will test this by parsing the text either manually or with a machine
TL3.4 Inputs: Same requirements as in the above 2

TL3.5 Outputs: The phrases with a BOOL value in them
TL3.6 Normal
TL3.7 Blackbox
TL3.8 Functional
TL3.9 Unit Test
TL3.10 Results: All BOOL phrases are set aside for further manual inspection

TL4.1 **LTL Acceptance Test 4**
TL4.2 This test will be an automated test comprising TL1, TL2 and TL3's requirements.
TL4.3 For this test we will test the same as the above, except in an automated manner
TL4.4 Inputs: Same requirements as above
TL4.5 Outputs: All of the processed sentences will be classified as "Yes" it is an LTL, or "No". There will be no "Maybe's" as we hope to sort these out by the end of TL3
TL4.6 Normal
TL4.7 Blackbox
TL4.8 Functional
TL4.9 Unit Test
TL4.10 Results: Yes or No if a phrase is an LTL

BM1.1 **Benchmark Test 1**
BM2.1 This test is also a manual test to see if the LTL acceptance value is too high
BM3.1 We will see if the LTL acceptance rate is too high. Per the JPL at CalTech, only >7% of phrases are temporal. From our smaller data set we've seen acceptance of up to 20%
BM4.1 Inputs: All LTL's classified as "YES"
BM5.1 Output: (LTL Yes's) / # of sentences => some percentage
BM6.1 Normal
BM7.1 Whitebox
BM8.1 Functional
BM9.1 Integration (are we classifying correctly?)
BM10.1 Results: We see if the LTL acceptance rate is deemed acceptable. It was upon manual acceptance

CK1.1 **Cohen's Kappa Test 1**
CK1.2 This test will test the level of agreement between two data sets
CK1.3 We will test this by computing Cohen's Kappa and using the benchmark values to assess the teams level of collective agreement from their manual classification of the requirements. This helps us develop a protocol of how to go about implementing a coded algorithm
CK1.4 Inputs: Manual Yes/No classification of LTL's
CK1.5 Output: A value from 0 to 1
CK1.6 Normal
CK1.7 Whitebox
CK1.8 Performance
CK1.9 Unit
CK1.10 Results: A level of high agreement between team members

LR1.1 **Independent Labeling Review 1**
LR1.2 This will be another manual test on a team member's temporal phrases and non temporal phrases
LR1.3 We will test this manually once again
LR1.4 Inputs: phrases reviewed by team member
LR1.5 Outputs: if the phrase is temporal or not
LR1.6 Normal
LR1.7 Whitebox
LR1.8 Functional
LR1.9 Unit
LR1.10 Results: will be a "peer review" test

TE1.1 **Training and Evaluation Test 1**
TE1.2 This test will utilize the ML algorithm and verify accurate labeling of the requirements.
TE1.3 10-fold cross-validation will be used, where the original sample is partitioned into 10 subsamples. The first 9 subsamples will be used to train the data and the final subsample will be used for evaluation.
TE1.4 Inputs: Documents/requirements from Dr. Niu.
TE1.5 Outputs: Precision, confusion matrix, other ML indicators.
TE1.6 Normal
TE1.7 Blackbox
TE1.8 Performance
TE1.9 Unit
TE1.10 Results: Check accuracy and precision in confusion matrix. Verify that these values indicate that our ML algorithm is correctly labeling the requirements.

AI1.1 **Algorithmic Integrity Test 1**
AI1.2 This test will validate the integrity of our ML algorithm.
AI1.3 Requirements provided by Dr. Niu (39, to be exact) will not be utilized in training and evaluation of our algorithm. As this is more data, it will verify the integrity of the algorithm.
AI1.4 Inputs: Documents/requirements from Dr. Niu.
AI1.5 Outputs: Precision, confusion matrix, other ML indicators.
AI1.6 Normal
AI1.7 Blackbox
AI1.8 Performance
AI1.9 Unit
AI1.10 Results: Check accuracy and precision in confusion matrix. Verify that these values indicate that our ML algorithm is correctly labeling the requirements.

QA1.1 **Algorithm Quality Assessment Test 1**
QA1.2 This test will be a quality benchmark of our algorithm
QA1.3 This test that confirms that our algorithm results align with the manual tests
QA1.4 Inputs: input into algorithm the manual results and the algorithm's results

QA1.5 Outputs: a value of agreeance, similar to Cohen's-Kappa Test 1 (CK1)
QA1.6 Normal
QA1.7 Blackbox
QA1.8 Unit
QA1.9 Integration
QA1.10 Results: Will output a level of agreeance with the manual tests and the algorithms results

**Test Case Matrix**

| | Normal / Abnormal | Blackbox / Whitebox | Functional / Performance | Unit / Integration |
|---|---|---|---|---|
| TL1 | Normal | Whitebox | Functional | Unit |
| TL2 | Normal | Whitebox | Functional | Unit |
| TL3 | Normal | Blackbox | Functional | Unit |
| TL4 | Normal | Blackbox | Functional | Unit |
| BM1 | Normal | Whitebox | Functional | Integration |
| CK1 | Normal | Whitebox | Performance | Unit |
| LR1 | Normal | Whitebox | Functional | Unit |
| TE1 | Normal | Blackbox | Performance | Unit |
| AI1 | Normal | Blackbox | Performance | Unit |
| QA1 | Normal | Blackbox | Performance | Integration |