****

TEXT processing algorithm FOR financial news analytics

**Student Name :** Soh Jun Jie

**Admin no. :** 1003346B  
**MP Group :** SOLY  
**Project ID :** FBI1215

**Project Title :** nTrader

# TABLE OF CONTENTS

1. Introduction 1

2. Background of Natural Language Processing 1

3. News Analytics NLP algorithm 2

Section 3.1 SOLY Text Analysis Algorithm 2

Section 3.2 NLP Algorithm Methodology & Techniques 3

4. Automating Financial Jargons Learning 9

5. Conclusion 10

6. Appendices 11

7. Bibliography 16

# 1. Introduction

The foundation features of Project nTrader is the ability to *spot future changes in market movement based on sentiment of financial news and also simplify technical financial jargons in it for better understandability*. It is also the basis in which other functionality of the system will depend upon, for example, to collect and archive accurate news sentiment analysis data, and to perform data analytics on these collected data. Essentially the system must rely on a text analysis algorithm to process natural language (the languages we all use to understand one another) in order for this to be possible. Such algorithm must be robust in order to produce reliable results.

The main challenges in creating an efficient and robust algorithm however for natural language processing (NLP) in the context of financial news and sometimes raw texts in general, lies in the ambiguity of these language, building of large enough dataset to support this algorithm, and the continuous expansion of new financial buzzwords and terms. This literature review will provides a brief introduction to NLP and also delves into NLP algorithms used by the industry now and the text analysis algorithm created by SOLY team. We aim to bridge the best of both, in which will allow our system to capture the semantic meaning of a phrase in context, as well as solving the above mentioned challenges through recommendations throughout this document.

# 2. Background of Natural Language Processing

The first formal research related to NLP-based problem solving started from 1950s to 1970s when publications concerning Machine Translation (MT) in the late 1940s sparked interests. The research aiming to materialize machine-based language translation adopted an over-simplified algorithm – using dictionary lookup for a word-for-word-based translation; semantic issues due to polysemy therefore arise and with slowdown in NLP research progress by 1966, the research was declared fail and government funding for NLP research stopped (Jones, 2001). Many NLP research activities had to terminate due to funding shortage, however basic issues like system architecture and processing methodologies for NLP were addressed; the emphasis in integrating Artificial Intelligence (AI) with some world knowledge to NLP, and creation of phrase-structure grammar technique which helps to convert natural language to machine-readable format serves as learning point (Lichtig, 2011).

The next phase of NLP research in the late 1970s to late 1980s however, sees the sudden rise in NLP research activities, owing largely to cheaper and better quality computational and linguistic resources i.e. parsers, grammatical tools, better database and computers (Bordes, Glorot, Weston, & Bengio, 2012). This results in many programs written specifically for NLP. For example, a chatbot program concept *LUNAR* was attempted in 1970s which have demonstrated to be able to answer questions in its domain with over 90% accuracy (Linckels & Meinel, 2011). Such projects motivated interests and funding in AIs and NLP research. Researchers had abandoned the previous concept of machine translation while starting to rivet at the linguistic of NLP, in particular grammatical logic computation, building various grammar types to annotate meaning and an increased focus in lexicon growth (Jones, 2001).

The surge in electronic materials after the late 1980s to 2000s then gave rise to new data source and testbeds for NLP, while bringing about a refocus in more complex researches, from message processing to text summarizing, and from documents’ abstract-only searching to NLP-based full-text searching of documents’ content (Jones, 2001). The NLP community began to develop probabilistic grammar tagging models for more accurate and robust NLP algorithm while developing logic to handle structure of sentences that are not as ideal. By 1987, the NLP community has begun experimenting speech recognition technology and in 1991 speech recognition and NLP was officially integrated with the help of DARPA Speech Recognition Workshop.

Today, developing and testing NLP applications are also much faster and easier as more advance computing resources becomes available and with years of NLP knowledge build up. NLP are now used for a wide range of applications area, including text summarizing, text sentiment analysis, language translation, and as spelling checkers. In the finance domain, NLP is gaining popularity in sentiment analysis of news and social media i.e. Twitter as traders add social media and news informatics as part of their trading strategies; research have shown that news and social media affects market psychology and mechanics thus affecting market prices (Ellis, 2012). For example, factors like Merger & Acquisition, mind-blowing sales figures, or joint venture agreement will cause a positive effect on stock prices. Similarly, news about a company going into bankrupt will have a negative effect on stock prices. However, the challenge in NLP now is to have comprehensive lexicon and statistical information to support accurate and robust processing.

# 3. News Analytics NLP Algorithm

This section describes SOLY text analysis algorithm and how the NLP methodology used by the industry to process text and derive meaning can be applied to strengthen our algorithm.

## 3.1 SOLY Text Analysis Algorithm

As the current text analysis algorithm used by SOLY to generate sentiment indicator, i.e. bull or bear, is based on only the news headline, it is allowed to be over-simplified.

Consider the following news headline which we scoped to select only those which have the word “stocks” as part of its headline:

1. European Stocks Decline as Moody’s Downgrades France
2. Hong Kong Stocks Drop, Reversing Gains, as Citic Pacific Slides
3. Chinese Stocks Fall to Seven-Week Low on FDI Drop

The type of the news that we scoped out all have articles’ headline following the recurring pattern of <Subject Stocks> <Direction> <Reason>, where <Subject Stocks> refers to the type of stocks in the headline e.g. European for the first sentence, Hong Kong for the second, and Chinese for the third. <Direction> refers to the price direction of the <Subject Stocks> and <Reason> refers to the factor that causes such price direction to the stocks.

SOLY algorithm derives the sentiment indicator by removing the <Reason> from the news headline. We does this by having a hardcoded list of ‘reason prepositions’ (e.g. on, as, before, amid) and stripped away the remaining text starting from the ‘reason prepositions’. This allows our algorithm to focus on the important part of the news headline. We derive the sentiment indicator then by searching for the <Direction> word, which we then match it with our knowledge base and assign the word with a sentiment value ranging from -1 to 1. -1 being bearish and 1 being bullish. Figure 1 and 2 in Appendices shows a snapshot of our database data which are used to capture news subject and directional factors in news headline.

However, this algorithm does not analyze words in context and will not work against news whose headline does not conforms to this format; this is even so if this algorithm is to be use in analyzing large amount of text while making meaning out of them. In order to introduce a robust NLP framework into SOLY text analysis algorithm, we recognize that sentences must be arranged into the <Subject Stocks> <Direction> <Reason> format or similar. Section 2.1 will focus on common NLP techniques and algorithm used by the industry and how they can help us achieve our algorithm requirements for robust news sentiment analysis.

## 3.2 NLP Algorithm Methodology & Techniques

We will review on open-text semantic parsers algorithm to format sentences into our desired <Subject Stocks> <Direction> <Reason> format or similar. As opposed to in-domain semantic parsers which are suited for single specific business domain, we chose to consider open-text (open-domain) semantic parsing as it can format any kind of natural language sentences of different depth of complexity while assigning an accurate meaning representation to text (Bordes, Glorot, Weston, & Bengio, 2012). The algorithm will be particularly suited for our financial jargon translation functionality where lexemes of a word vary in meaning when put into different context. This will allow us to derive with accuracy the meaning of financial jargons in context. Sentiment indicator can also be based on this algorithm with the formatted sentences output. Another reason for selecting open-text semantic parser is also because in-domain semantic parser requires highly trained financial domain training corpus which is beyond our development scope and would be too costly to implement, due to the time and effort. A domain specific training corpus are generally commercialized versus open-domain training corpus which are free for download by researchers i.e. Brown corpus.

NLP has 3 techniques to help us derive meaning representation from an input sentence – (1) Part-Of-Speech (POS) tagging, (2) Text Chunking & Named Entity Recognition, and (3) Semantic Role Labeling.

1. Part-Of-Speech (POS) Tagging

POS tagging of sentence structure is essential in meaning representation due to the fact that word of several POS classifications takes different meaning when place into context. Consider the following example:

1. The investors bull on Apple Inc.
2. Price of bull will increase next month.

Both sentences used the word bull but convey different meaning; the ‘bull’ in sentence 1 act as a verb while in sentence 2, ‘bull’ is a noun. By POS tagging all of the words in a sentence (only the word ‘bull’ for below, VB for verb and NN for noun), certain level of lexical ambiguity can be removed. We suggest using Penn Treebank tagset of 36 POS tags or better to tag words as it is less ambiguous versus the basic POS tagset of only noun, verb, adjective and adverb. We will use Penn Treebank tagset this document’s examples (Refer to appendices Table 1).

1. The investors bull/VB on Apple Inc.
2. Price of bull/NN will increase next month.

Note that there are various methods for POS tagging such as the Unigram tagger, Hidden Markov Model (HMM), and transformation-based tagging. However, we will only elaborate more on the *transformation-based tagging approach as it considers semantic meaning of words and is much more scalable.*

In summary however, the Unigram tagger approach is trained based on a training corpus. Therefore a particular tag is assigned based on the most common usage of that word and may not be in context. For example, the word ‘bull’ (animal) is more commonly used as a noun than a verb (e.g. bull in stock prices). Unigram tagger approach will assign the ‘NN’ noun tag for the bull word regardless of the context (Hasan, UzZaman, & Khan, n.d.).

HMM approach on the other hand is a probability-based approach that assigns tag that maximizes the following formula:

P(Word|Tag) \* P(Tag|Previous n tags)

Take for example the word ‘bull’ has a 0.78 probability of coming with a noun based on the training corpus, P(Word|Tag)will be 0.78. P(Tag|Previous n tags) represents the probability that the assigned tag comes with a tag that is *n* number of times before or after it, which is also based on a training corpus. We will not use this approach as it does not consider linguistic rules and machine learning of linguist rules versus the transformation-based approach.

**More on Transformation-based POS tagging**



Illustration 1 - Model of Transformation-based Error-Driven learning algorithm

Illustration 1 illustrates the Transformation-based process.

Step 1: Assign POS Tags

With an untagged input sentence, POS tags will be assigned to each word in the sentence based on the most common usage of that word through a training corpus. There will also be transformation rules database which transforms the previously assigned tag based on the preceding or succeeding tag. A transformation rule such as “change the NN (noun) tag to VB (verb) tag if the preceding tag is TO (TO represents the word ‘to’)” is an example.

Step 2: Archive Sentence to Training Corpus

The sentence is now said to be POS tagged. The supervisor of the POS tagging system can manually tagged the same sentence (called *TRUTH*) and archives it to the training corpus.

Step 3: Machine learning transformation-based rules

The *TRUTH* and the tagged sentence will become input into the Machine Learning Algorithm to output a transformation rule can makes the tagged sentence more resembles the *TRUTH*. This will dynamically grow the transformation rule database.

Following the above methodology for the financial jargon translation and news sentiment analysis, we will only process sentences that have financial jargons or important keywords necessary for making decisions in trading. This will remove associated performance issue with processing unnecessary sentences.

As SOLY does not have its own training corpus to pre-assign POS tags, we plan to either leverage on Brown Corpus to assign basic POS tags or simply hardcode a sufficient list of values to pre-assign basic words such as prepositions, symbols, pronouns and determiners. Our core logic is to first rely on a set of to-be-developed transformation rules to transform incorrectly assigned POS tags which is based on linguistic rules instead of a training corpus, so POS tagging will be more accurate. The transformation rules will be built slowly. The learning algorithm may not be in the scope due to time constraint of 10 weeks.

1. Text Chunking & Named Entity Recognition

The concept of Text chunking & Name Entity Recognition is to group individual words into a single logical phrase or recognizable entity i.e. noun phrase. Take for example, ‘television program’ is a single entity and is a noun phrase. It should not be treated as two individual words or entities.

The following table shows the different composition of a noun phrase and how we can chunk words into noun phrase using regular expression to detect order of POS tags (Zhu, n.d.).

|  |  |
| --- | --- |
| **Composition of noun phrase** | **Regular expression pattern** |
| A possible determiner (DT) or possessive pronoun (POS), followed by an optional Cardinal Number (CD), followed by any number of adjectives (JJ), and then a noun (NN) | (DT|POS|PP$|WP$)?\s+(CD)?\s+(JJ)\*\s+(NN) |
| Sequence of proper nouns (NP) | (NP+) |
| Sequence of nouns (NN) | (NN+) |

Words that are chunked into noun phrase will be denoted by the tag ‘NNP’. From here, financial jargons can be identified based on its POS tag. For example if the word ‘bull’ is tagged as a verb, we can identify it as a financial jargon. However as the word ‘bull’ can also denote many other sense of meaning in context, the Semantic Role Labeling process is the crucial step to disambiguate the word.

1. Semantic Role Labeling (SRL)

SRL involves assigning roles to words and sometimes formatting of a sentence for machine readability, meaning representation, and syntactic analysis. To represent meaning, machine is often interested in the who, what, when, where, how, and whom of a sentence (Yih & Toutanova, 2007).

To perform SRL, sentences must be broken down into different types of phrases: noun phrase, prepositional phrase, verb phrase. This will be broken down in a way that resembles a parse tree through a process called syntactic analysis (see appendices fig 3 for example). The text chunking step would have already identified the noun phrases. To detect and group prepositional phrases and verb phrases we will use the regular expression shown in the following (Blog::Quibb, 2010).

|  |  |
| --- | --- |
| **Composition of prepositional phrase (INP)** | **Regular expression pattern** |
| Preposition (IN) follow by a noun phrase (NNP) or noun (NN). | (IN)\s+(NNP|NN) |
| **Composition of verb phrase (VBP)** | **Regular expression pattern** |
| A verb (VB) followed by any number of modifier words and then a noun phrase (NNP) or prepositional phrase (INP). For this case, we assume modifier word to be any ordinary word. | (VB)\s+.\*(NNP|INP)+? |

For the sake of financial news sentiment analysis and financial jargon translation, we will only perform syntactic analysis on sentences which meets our POS tagging requirements – sentences having important keywords or financial jargons, or if the sentence having an explicit directional factor which is outlined in our database (Appendices Fig 1).

Following syntactic analysis, we will assign semantic roles to the phrases. Framenet is a lexical database having knowledge base in usage of verbs in sentences. It has pre-defined semantic roles for different group of verbs synset (synonym set). We will use Framenet *Motion Directional* frame structure for our semantic labeling. Please refer to appendices Table 2 for the list of semantic roles we will use. To do the labeling, we will study on the placement of noun phrases, prepositional phrases, verb phrases, punctuations, and conjunctions. Take the below input sentence from *Appendices: Fig 3* as example, which is the output of our text chunking step (note that not all of the words below are POS tagged for the sake of readability; our actual algorithm will have all of the words POS tagged).

1. [NNP The S&P 500]
2. [VBP rallied/VBD]
3. [NNP 1.3/CD percent/NN]
4. [INP on/IN]
5. [NNP the last day of the week] [SYM ,]
6. [VBP posting/VBG]
7. [NNP the best post-Thanksgiving performance]
8. [INP since/IN] [2007] [SYM ,]
9. [INP as/IN]
10. [NNP the American holiday shopping season]
11. [VBP began/VBD] [SYM .]

We will identify noun phrases (Tagged as *NNP*) and words without a phrase tag (i.e. [2007] in line 8) as a list of arguments that will take on these semantic roles. We will arrange them in the manner shown in the following (Santos & MilidiGu, 2012). ‘Args’ stands for argument and ‘V’ stands for verb. Note that there are also other argument types that will be considered besides ‘V’ and ‘Args’ but are not shown in the following sample (Refer to Appendices: Table 3 for the list of argument types).

[Args1 The S&P 500] [V rallied/VBD] [Args2 1.3/CD percent/NN] on [Args3 the last day of the week], [V posting/VBG] [Args4 the best post-Thanksgiving performance] since [Args5 2007], as [Args6 the American holiday shopping season] [V began/VBD].

Following that, we will identify related events and map events caused by the subject (noun phrases that uses the verb). We identify events which are also verb phrases using regular expression, and replace the text with a simplified form from the sentence to help in parsing unneeded text. After the events are identified, we will have the following results

|  |  |
| --- | --- |
| **Events identified** | * *[Evts1] -> [V rallied/VBD] [Args2 1.3/CD percent/NN] on [Args3 the last day of the week]* * *[Evts2] -> [V posting/VBG] [Args4 the best post-Thanksgiving performance] since [Args5 2007]* * *[Evts3] -> [V began/VBD]* |
| **Output sentence** | * *[Args1 The S&P 500] [Evts1], [Evts2], as [Args6 the American holiday shopping season] [Evts3].* |

Following the above, will then map the subject(s) and events as follow.

|  |  |  |  |
| --- | --- | --- | --- |
| **Subject** | **Events** | **Event description** | **Mapping logic** |
| [Args1 The S&P 500] | [V rallied/VBD] | * [Args2 1.3/CD percent/NN] * on [Args3 the last day of the week] | Verb phrase or consecutive Verb phrase delimited by comma position after a noun phrase |
| [Args1 The S&P 500] | [V posting/VBG] | * [Args4 the best post-Thanksgiving performance] * since [Args5 2007] | Verb phrase or consecutive Verb phrase delimited by comma position after a noun phrase |
| [Args6 the American holiday shopping season] | [V began/VBD] | NIL | Verb phrase or consecutive Verb phrase delimited by comma position after a noun phrase |

After the above step we can disambiguate words that may be financial jargons base on the type of events the word is located at, and study the relationship between the *event* and the arguments in the *event description* column and compare them to our search criteria which is tagged to each of our pre-identified financial jargons. Take the sentence “The investors bull on Apple shares” as example. The subject would be [Args1 The investors], the event would be [V bull] and event description would be on [Args2 Apple shares]. Our search criteria for the financial jargon ‘bull’ as stored in our database would then be “shares”, “stocks”, “Inc. ”, “Co. ”, “Company”. If we managed to identify any of these predefined criteria in the *event description*, we can assume the word is a financial jargon and perform jargon translation.

The next and last step would be mapping the subject and components in these events to semantic roles. We will map semantic roles based on the type of content in the phrases and the positions of prepositions and conjunctions. Please refer to Appendices: Table 2 for meaning of the semantic roles, the table in the following page will show the SRL logic.

|  |  |  |
| --- | --- | --- |
| **Phrases** | **Semantic Role** | **Labelling Logic** |
| [Args1 The S&P 500] | Theme | Find the noun phrase that triggers a directional event. We identify ‘rallied’ as a direction factor through our m0001\_direction\_keys table (Appendices: Fig1) which will store rallied as a directional factor. |
| [Args2 1.3/CD percent/NN] | Distance | Find the phrase that has a Cardinal Number and a Noun after the position of the direction verb ‘rallied’. The Cardinal Number with a Noun word is a measurement of the noun word. |
| on [Args3 the last day of the week] | Time | Find the phrase that has a preposition word followed by words that give clues to date i.e. day, week, year, date formatting. |
| [Args4 the best post-Thanksgiving performance] | Degree | Find the modification keywords in the noun phrase. Modification keywords can also be found in our m0001\_direction\_keys table which is marked with a ‘Modification’ type; direction keywords are marked as ‘Direction’. |
| since [Args5 2007] | Time | Find the phrase that has a preposition word followed by words that give clues to date i.e. day, week, year, date formatting. The year format in this case. |
| as [Args6 the American holiday shopping season] [V began/VBD] | Explanation | Find the preposition that give clues to reason. E.g. ‘as’, ‘after’, ‘amid’. The preposition must be position before a noun phrase. The phrase after the preposition until before a valid punctuation will take the Explanation semantic role. |

The SRL process is completed. By tagging these phrases with semantic roles, our application can define more information about events related to the financial instrument in a sentence, in this case, S&P 500. We will incorporate the algorithm with SOLY existing text analysis algorithm which currently does not take linguistic rules into account. For our system functionality, this will have the advantage of correctly identifying words in context for financial jargon translation, and more detailed reporting about financial instrument status will the tagging of semantic roles.

# 4. Automating Financial Jargons Learning

As with the continuous expansion of new financial buzzwords and jargons, keeping track of it will be a tedious process. To simplify the process, we recommend an implementation of machine learning algorithm that reduces manual jargon tracking and database maintenance of these jargons, when financial jargons are coined or are changed. Machine learning involves the programme to make improved decision-making based on new data and we have considered this technique in the recommended transformation-based POS tagging model. We will apply this technique by leveraging on *The Financial Times Lexicon* (http://lexicon.ft.com/) for their lists of financial jargons, which the site has already categorised methodically, which we can then browse the site using php language to search for new financial jargons that are recently added as well as any terms where the definition has been modified. SOLY can then automatically updates its financial jargon database by detecting these changes. This algorithm can be executed on a daily basis by scheduling our server (Amazon EC2 server) to do so. SOLY will also give users the option to suggest financial jargons. This way, the financial jargon translation functionality will be more self-sustainable and scalable.

# 5. Conclusion

In this paper we have looked at some of the challenges in Natural Language Processing and the constraints of the text analysis algorithm used by SOLY. The main challenge amongst others is that natural language is often ambiguous when placed in multiple contexts. Overcoming these challenges, the paper presented another approach in text analysis by demonstrating three NLP techniques introduced by industry professionals which we then improvised to our algorithm.

1. Part Of Speech tagging: We presented the transformation-based POS tagging approach which will use a model that is dependent on linguistic and grammatical rules rather than a training corpus. We believe this approach will enhance accuracy of POS tagging as far as grammatical syntax are concern. Only sentences containing keywords predefined by us will use this algorithm.
2. Text Chunking and Named Entity Recognition: We demonstrated the used of regular expression pattern matching to identify noun phrases.
3. Semantic Role Labelling: We have shown how regular expression again can identify prepositional phrases and verb phrases which we then map their components into a set of arguments. We also created mapping logic and labelling logic which will allow our program to capture semantic information and meaning of jargons of an input sentence.

With the above three techniques implemented, the final system will be able to interpret natural language sentences that meets a set of constraints (i.e. sentences containing keywords). While our ambition for the text analysis algorithm is to interpret all kinds of natural language sentences for more value added user reporting such as financial instrument summary, this feature may not be available in the near-term due to the algorithm complexity. Instead, SOLY will redirect future effort to more accomplishable algorithms such as financial jargons machine learning.

# 6. Appendices

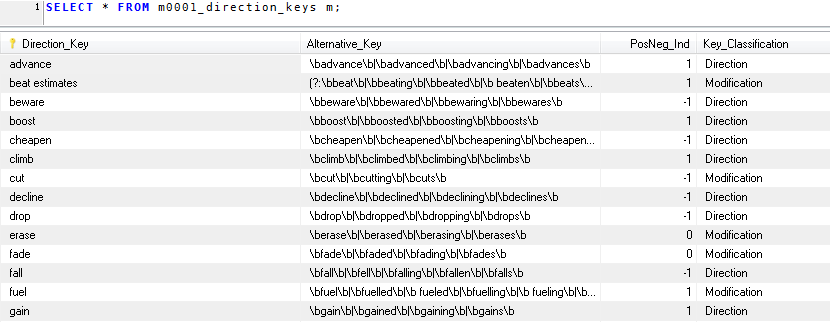


Fig1. Database snapshot of SOLY’s m0001\_direction\_keys table showing directional factors

The m0001\_direction\_keys table helps to identify the direction key in a sentence. The Alternative\_Key column is a regular expression pattern that matches for lexemes of a direction key lemma. A PosNeg\_Ind column indicate the sentiment value of the lemma word in the column Direction\_Key. We use this indicator to compute the sentiment of a sentence that our algorithm process.

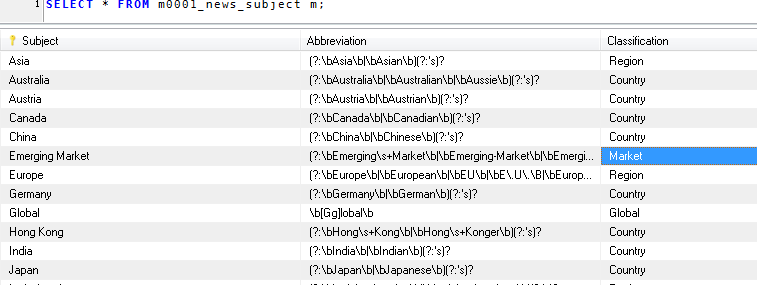


Fig2. Database snapshot of SOLY’s m0001\_news\_subject table showing news subject

The m0001\_news\_subject table helps to identify the subject in a sentence that users are concern about. This subject could be a business entity, financial instrument, a country or a region. The subject are capture in the table ‘Subject’ column. The Abbreviation column is a regular expression pattern that matches for lexemes of a Subject lemma. This way we can identify a subject of concern in a sentence.



Fig 3: Syntactic Analysis Parse tree

The above shows the syntactic breakdown of a sentence to three types of phrases – noun phrases, prepositional phrases and verb phrases. Not all words/phrases in a sentence have to be classified as one of these three phrases types, some examples would be modal verbs and conjunctions.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **POS Tag** | **Description** | **Examples** |
| 1 | CC | Coordinating conjunction | And |
| 2 | CD | Cardinal number | 1, third |
| 3 | DT | Determiner | The |
| 4 | EX | Existential there | there is |
| 5 | FW | Foreign word | d'hoevre |
| 6 | IN | Preposition or subordinating conjunction | in, of, like |
| 7 | JJ | Adjective | Green |
| 8 | JJR | Adjective, comparative | Greener |
| 9 | JJS | Adjective, superlative | Greenest |
| 10 | LS | List item marker | 1) |
| 11 | MD | Modal | could, will |
| 12 | NN | Noun, singular or mass | Table |
| 13 | NNS | Noun, plural | Tables |
| 14 | NP | Proper noun, singular | John |
| 15 | NPS | Proper noun, plural | Vikings |
| 16 | PDT | Predeterminer | both the boys |
| 17 | POS | Possessive ending | friend's |
| 18 | PP | Personal pronoun | I, he, it |
| 19 | PP$ | Possessive pronoun | my, his |
| 20 | RB | Adverb | however, usually, naturally, here, good |
| 21 | RBR | Adverb, comparative | Better |
| 22 | RBS | Adverb, superlative | Best |
| 23 | RP | Particle | give up |
| 24 | SYM | Symbol | $, % |
| 25 | TO | to | to go, to him |
| 26 | UH | Interjection | uhhuhhuhh |
| 27 | VB | Verb, base form | Take |
| 28 | VBD | Verb, past tense | Took |
| 29 | VBG | Verb, gerund or present participle | taking |
| 30 | VBN | Verb, past participle | taken |
| 31 | VBP | Verb, non-3rd person singular present | Take |
| 32 | VBZ | Verb, 3rd person singular present | takes |
| 33 | WDT | Wh-determiner | which |
| 34 | WP | Wh-pronoun | who, what |
| 35 | WP$ | Possessive wh-pronoun | whose |
| 36 | WRB | Wh-adverb | where, when |

Table 1: Penn Treebank POS Tagset

Source: <http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/CQP-HTMLDemo/PennTreebankTS.html>

|  |  |  |  |
| --- | --- | --- | --- |
| **Semantic Roles** | **Abbreviation** | **Description** | **Example** |
| Theme | thm | The entity that is affected by the direction | *Apple Stocks* advances to three-weeks high |
| Direction | dir | The motion movement taken by the theme | Apple Stocks *advances* to three-weeks high |
| Goal | gl | The location the theme ends up in | USD drops from 1.25 to *1.20* against SGD |
| Source | src | The initial location the theme occupies before the motion movement | USD drops from *1.25* to 1.20 against SGD |
| Degree | dgr | The extent of the direction motion affecting the theme | Asian stocks rose to *record-high* since 2008 |
| Distance | dis | The difference between the source and goal | S&P 500 drop by *1000 points* to 2900 points |
| Duration | dur | Length of time which the direction motion occurs | Gold continues to rally for *3 weeks* straight |
| Explanation | exp | The reason that causes such direction motion | Emerging Stocks Gain for Fifth Day as *Taiwan Shares Rally* |
| Time | tim | The time which the motion movement occurs | U.S. stocks have best weekly rally *Since June* on Budget |

Table 2: Sematic role labels for motion directional frame sentences

Source: <https://framenet2.icsi.berkeley.edu/fnReports/data/frameIndex.xml?frame=Motion_directional>

|  |  |  |
| --- | --- | --- |
| **Argument type** | **To represent** | **Description** |
| V | Verb | ‘V’ stands for ‘Verb’. Verb conveys an action. We interpret a noun phrase that position before a verb as a noun that uses the verb action. |
| Args | Noun phrases | ‘Args’ stands for a standard ‘Argument’. They will take on a representation of a noun phrase to show interaction with verb. |
| AM-Neg | ‘Not’ and ‘n’t’ | ‘AM-Neg’ inverts the meaning of a modal verb that lies before it. For example the word “wouldn’t” will translate to “would [AM-Neg n’t]”. [AM-Neg n’t] will invert the meaning of the modal verb before it. |
| AM-Mod | Modal verbs | Modal verbs such as ‘would’, ‘can’, ‘may’ will be tagged as ‘AM-MOD’ |

Table 3: argument types for Semantic Role Labelling

Source: Entropy Guided Transformation Learning [Ebook] by Santos, C. N., & MilidiGu, R. L.

# 7. Bibliography

Blog::Quibb. (2010, January 27). NLTK regular expression parser (RegexpParser). Retrieved November 25, 2012, from http://blog.quibb.org/2010/01/nltk-regular-expression-parser-regexpparser/

Bordes, A., Glorot, X., Weston, J., & Bengio, Y. (2012, April 26). Joint learning of words and meaning representations. Retrieved November 20, 2012, from http://jmlr.csail.mit.edu/proceedings/papers/v22/bordes12/bordes12.pdf

Brill, E. (1994). Some advances in transformation-based part of speech tagging. Retrieved November 24, 2012, from http://www.aaai.org/Papers/AAAI/1994/AAAI94-110.pdf

Ellis, E. (2012, August 10). Seth McGuire on social media and the stock market. Retrieved November 22, 2012, from http://blog.gnip.com/social-media-and-the-stock-market/

Hasan, F. M., UzZaman, N., & Khan, M. (n.d.). Comparison of different POS tagging techniques (N-Gram, HMM and Brill’s tagger) for Bangla. Retrieved November 23, 2012, from http://www.panl10n.net/english/final%20reports/pdf%20files/Bangladesh/BAN08.pdf

Jones, K. S. (2001, October). Natural language processing: A historical review. Retrieved November 20, 2012, from http://www.cl.cam.ac.uk/archive/ksj21/histdw4.pdf

Lichtig, R. (2011, December). The history of natural language processing. Retrieved November 21, 2012, from http://www.ieeeghn.org/wiki/index.php/The\_History\_of\_Natural\_Language\_Processing

Linckels, S., & Meinel, C. (2011, April 12). E-librarian service: User-friendly semantic search in digital libraries. Retrieved November 21, 2012, from http://books.google.com.sg/books?id=81rtxgnq4F4C&printsec=frontcover&source=gbs\_ge\_summary\_r&cad=0#v=onepage&q&f=false

Santos, C. N., & MilidiGu, R. L. (2012). *Entropy guided transformation learnin.* Dordrecht: Springer.

Yih, S. W.-t., & Toutanova, K. (2007, July 23). Automatic semantic role labeling. Retrieved November 25, 2012, from http://research.microsoft.com/pubs/101987/srl-tutorial-aaai-07-0723.pdf

Zhu, D. (n.d.). Text chunking using NLTK. Retrieved November 24, 2012, from http://www.eecis.udel.edu/~trnka/CISC889-11S/lectures/dongqing-chunking.pdf