

# Estimating Credibility of News Authors from their WIKI Validated Predictions

Navya Yarrabelly  
DSAC, IIIT Hyderabad  
yarrabelly.navya@research.iiit.ac.in

Kamalakar Karlapalem  
DSAC, IIIT Hyderabad  
kamal@iiit.ac.in

## Abstract

In this paper, we consider a set of articles or reports by journalists or others, wherein they predict or promise something about future. The problem we approach is determining the credibility of the authors based on the predictions coming out to be true. The two specific problems we address are extracting the predictions from the articles and annotating with various prediction attributes. And then we determine the truth of these predictions, using Wikipedia as a credible source to extract relevant facts which can ascertain the validity of the predictions. We proposed and built an end to end system for *automated predictions validation*(APV) by extracting future speculations and predictions from news articles and social media. We considered 28 news articles and extracted 97 predictions from these articles and the range of credibility scores(F-scores) for these articles are (0.57-0.71).

## 1 Introduction

In newspaper articles, many journalists evaluate the current state of affairs and predict possible future scenarios. [6] estimates from their investigations that nearly one-third of news articles contain predictive statements. Therefore, it is imperative to determine the passages, sentences and phrases of the news articles that predict the future scenarios. A person well versed

with reading articles can easily determine predictability aspects of a news article and over time has some assurance about which articles or news agencies correctly predict some of the future scenarios. It is important and necessary, therefore, to enhance our ability to computationally determine the credibility of journalists based on their ability to predict the future scenarios correctly. As a step towards this direction, we take up the automatic verification of predictive statements against facts collected from credible information sources. This task of machine reading at scale has the difficulties of relevant article retrieval (finding the significant facts) with that of machine perception of content (entailment of predictions from facts).

Consider the following prediction published on date ‘d’.

**Example:** The Reserve Bank of India may lower the economic growth projection for 2017-18 to 6.7 per cent later this month, from its August forecast of 7.3 per cent, in view of issues with GST implementation and lower kharif output estimates. In the above predictive sentence, we have to precisely extract and validate only the predictive part “*The Reserve Bank of India may lower the economic growth projection for 2017-18 to 6.7 per cent*”, “*in view of issues with GST implementation and lower kharif output estimates.*” is the premise on which the prediction is made and “*from its August forecast of 7.3 per cent*” is a supporting clause. The reference future date for this prediction “later this month” is translated to actual date ‘d+30’. The facts relevant to the predictive part, which are published after the target date ‘d+30’, are extracted to determine the entailment relation from fact to prediction.

**Contributions:** The main contributions of the approach proposed are

(1) To translate predictions to structured queries, we annotate the predictions with a wide range of attributes(in Table 1). This can further be used by an IR system to retrieve predictions made in reference to a future time period, targeting an event etc.

---

Copyright © 2018 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors.

In: D. Albakour, D. Corney, J. Gonzalo, M. Martinez, B. Poblete, A. Vlachos (eds.): Proceedings of the NewsIR'18 Workshop at ECIR, Grenoble, France, 26-March-2018, published at <http://ceur-ws.org>

(2) We also report a timeline story of its relevant facts and analysis, and the fact sources confirming the truth of the predictions. News IR systems can also come up with recommendations or follow up links for an article read, based on the predictive attributes and from the timeline of facts extracted.

(3) We propose an approach to tackle open-domain prediction validation using Wikipedia as the unique knowledge source.

## 2 Related Work

Research has in the past focused on how to answer questions but has not devoted attention to discerning the accuracy of the predictions/promises made. To the best of our knowledge [5] is the only work which focused on the estimation of validity of predictions, by calculating cosine similarity between predicted news and the relevant events that actually occurred. We offer semantic and syntactic analysis based on the structure of relation triplets in a predictive sentence and incorporated domain-specific knowledge into the system. Also, their retrieval model is limited to topics contained by predictions (manually collected). Though applications on future information retrieval have been studied by a number of researchers, study on the problem of validating predictions from Natural Language Understanding perspective is limited. [7] presents a search engine for future and Past events relevant to a users query. [3] automatically generates summaries of future events related to queries. Their methods rely on extracting and processing statements containing future temporal references. [6] retrieves and ranks predictions that are relevant to a news article using features: term similarity, entity-based similarity, topic similarity, and temporal similarity.

### Relevance to Fact Checking and QA Systems

To some extent our problem can be compared with the Fact Checking and Question Answering systems. Though research has been done on the truth assessment of fact statements relying on iterative peer voting, leveraging language to infer accuracy of fact candidates has only started. [14] calculates the credibility of an uncertain fact by comparing other related facts. Fact validity is estimated by the co-occurrence degree of the doubt object and predicate by relying on page counts for web queries.[4], [8] proposed to convert a fact-checking question into a set of factoid-style questions and validated the answers against those retrieved by Factoid Question Answering systems Our problem differs from existing fact checking systems and question answering systems in its retrieval problem, as we only have to validate the predictive part of a sentence and retrieve the relevant facts which occurred within the implicit temporal constraints imposed.

## 3 Predictions

### 3.1 Predictions Extraction :

From each article, we annotate the sentences as predictive or factual using the implementation from [15]. It also identifies the predictive phrase in the prediction and resolves the scope of the prediction in a complex sentence.

### 3.2 Semantic Graph Model for Predictive Sentence Simplification

News articles often contain long and syntactically complex sentences with relevant dependent relations spanning over various clauses. It is required to determine constituents that commonly supply no more than contextual background information. Inspired by the work of sentence simplification using relation graph<sup>1</sup> and syntactic sub-structures [11, 1], we followed a syntax based sentence simplification approach to determine such constituents and to annotate predictions with various attributes. We constructed a Triplet-Level Semantic Graph Model (TLSGM) which has relation-triplets as vertices and the semantic relationships between the triplets govern the edges in the graph. From the TLSGM, we identified core triplets of the predictive part of the sentence and dis-embedded other peripheral triplets w.r.t the head predictive phrase extracted in Section 3.1. Then only these core triplets are validated to determine the accuracy of the prediction.

**Vertices:** Vertices in the TLSGM represent (subject, predicate phrase, object) relation triplets extracted from the prediction.

**Edges:** An edge between two nodes  $N_1 \rightarrow N_2$  represents the semantic relation of node  $N_2$  w.r.t node  $N_1$ . Edges can be formed either from the subject or object of a node to another node describing/modifying the noun phrase of subject/object, following the rules for noun descriptors. While edges formed from a predicate to another node follow verb descriptor rules given below. We illustrate the descriptor rules using example sentences given below.

1. Example 1 : Mary Kom, who won Bronze at London Olympics, still has a fifty-fifty chance of gaining a wildcard entry to the 2016 Rio Olympics. (Mary Kom, has, fifty-fifty chance) is the head predictive triplet (H).
2. Example 2 : The Reserve Bank of India is likely to leave interest rates unchanged in order to keep inflation rate controlled.

### Rules for Noun Descriptors

Modifiers and Dependents of the head of the noun

<sup>1</sup><https://github.com/Lambda-3/Graphene>

phrase of either the subject or object of a triplet are discussed below, categorized by the dependency relations.

**acl:relcl, appos** : A relative clause modifier from the head noun of an NP to the head of a relative clause. The clause introduced by this dependency only gives additional information on the noun phrase and does not remark about the future predictive action, which is our focus of interest. Example 1 has relation `acl:relcl(Sindhu, won)` from the subject of node H. And the edge between H and  $N_2$ : (Mary Kom, won, Bronze at London Olympics ) is only an additional descriptor of H. Node  $N_2$  and its edges are pruned from the graph.

**acl** : An adjectival clause introduced by a Noun.

- If the dependent is a verb, and it has no subject, it takes the object of the governor. Example 1 has a relation `acl(chance,gaining)` from the object of H. And the edge between H and  $N_2$  : (fifty-fifty chance, gaining, wildcard entry to the 2016 Rio Olympics) further specifies the predictive action of  $N_1$  and hence node  $N_2$  is retained in the graph.

- If the dependent is an adjective, it will only describe the subject/object. This relation is also used for optional depictives to modify the nominal of which it provides a secondary predication. Example 2 has a relation `acl(rates, unchanged)` from the object of H. And the edge between H and  $N_2$  : (interest rates, unchanged) acts as a qualifier reference for the entities contained in the prediction.

#### Rules for Verb Descriptors

**xcomp** : An open clausal complement (xcomp) of a VP, without its own subject, whose reference is determined by an external subject.

- If the governor of the relation contains an object of its own, the clause introduced by xcomp provides attributes to the relation contained by the governor predicate and acts as a purpose or consequence clause. Ex : Microsoft share values may go down by 10 dollars to give space to the new iPhone launch. We create an edge (Microsoft share values, may go down, by 10 dollars) -<sub>i</sub> (,give,space to the new iPhone launch), governed by the relation `xcomp(go, give)`.

- If the governor of the relation does not contain an object, the dependent predicate modifies the head predicate. We modify the predicate of the current node to include the dependent predicate connected by xcomp relation. Example 2 has a relation `xcomp(likely, leave)`. We modify H to (The Reserve Bank of India, is likely to leave , interest rates unchanged ).

**ccomp for a verb** : A clausal complement of a verb is a dependent clause with an internal subject which functions like an object of the verb, or adjective. The clause introduced further describes the future course of action referred by the governor predicate P. Ex: Modi promised that Indian GDP growth rate would cross

8% this year has a relation `ccomp(promised, cross)`, which adds an edge from (Modi, promised,) to (GDP growth rate, would cross, 8%) .

**advcl** : An adverbial clause modifier of a VP or S is a clause modifying the verb to introduce either a temporal, consequence, conditional or purpose clause and adds specificity to the head clause. Example 2 has a relation `advcl(leave, keep)` which adds an edge from H to triplet  $N_2$  : (RBI, to keep , inflation rate controlled). The validity of the predictive sentence should be determined regardless of the state of truth of the purpose/conditional clause. Hence the node  $N_2$  and its edges are discarded from the graph.

### 3.3 Prediction Attributes

Each Node in TLSGM is further classified and labeled with reference to the root node i.e head prediction node of the graph. We have determined the characteristics of following constituents, using a number of syntactic features (dependency relation types, constituency-based parse trees as well as POS and NER labels). **Attributes** : (Action; Event; Event location; Event Time; Purpose / Consequence of predictive action; Premise; Conditional clause; Qualifier Reference which adds specificity attributes of the entities involved in the prediction; Numeric Quantifier Reference; Certainty Perspective to isolate predictive stances taken by an author from third party's voices that are presented by the author).

## 4 Extracting Relevant facts

In the following section, we describe our system for Automatic Prediction Validation (APV) which consists of three components: (1) Keyword selection module to select keywords specific to the predictive part, dis-embedding the linguistic peripheral clauses identified in section 3.2 (2) the Document Retriever module for finding facts relevant to the prediction and (3) a machine comprehension model, Document Reader, for ascertaining the accuracy of predictions from a small collection of relevant facts.

### 4.1 Keyword Selection

Obtaining the pertinent facts relevant to the prediction is in itself a complicated problem to solve. Predictions have event and temporal based constraints, clausal complements, appositives, relative clauses etc. to add specificity or modify the action of an event. To overcome the problem of query drift introduced by these clauses, we further dis-embed keywords expressing the time constraints, premise clauses, certainty perspective (annotated in Section 3.2) and the speculative words used. We identify the headword of the

predictive phrase and used a rule based approach so that the predictive sentence fragments can be detected and to select keywords pertaining to the predictive action and its attributes in the sentence. Let  $K$  be the set of relation triplets, we add the head vertex of the graph (TLSG) to  $K$  and recursively add selected nodes from its edges to  $K$ . We select nodes with edge labels corresponding to Action, Event, Qualifier and Quantifier References as described in Table 1. We then give proximity queries where subject, predicate and object occur within a window of 7 words. We further expand the query set iteratively by adding purpose clauses and expand keywords in a query with their synonyms.

Example: For the predictive sentence “Lizzie Armitstead is predicted to win gold medal in cycling road race at the Rio Olympics. ”

Query : (Lizzie Armitstead  $\sim$  win  $\sim$  gold medal) OR (Lizzie Armitstead  $\sim$  win  $\sim$  cycling road race) OR (Lizzie Armitstead  $\sim$  win  $\sim$  Rio Olympics. )

## 4.2 Candidate Relevant Facts Extraction From Wikipedia

To extract pertinent facts which can ascertain the accuracy of the predictions, we used Wikipedia as a knowledge source. Wikipedia’s publicly available apis<sup>2</sup> to access revision history of each article and its up-to-date knowledge marked with timestamps makes it a reliable source for event-based prediction validation. We used tagme<sup>3</sup> as a semantic interpreter that maps fragments of natural language text into a weighted sequence of Wikipedia concepts relevant to the input. Using the query set in the above step(Section 4.1), we extracted the top 50 documents, from a local Lucene index of Wikipedia English dump. To further extract the relevant snippet from the article, we only included the article content with revision dates occurring within the time-window referenced by temporal constraints extracted for the validity of the prediction. We used the word2vec python implementation of Gensim [13] using Wikipedia as a corpus for generating embeddings to represent contextual term vectors. Inspired from [10], we adapted Zero Filter, Terms filter, Exact Sequence Filter, Normalization Filter, N-grams Filter, Density Filter to extract and sort the relevant candidate facts from the retrieved articles. Additionally, we implemented the following filters

- Distance filter : Assigns a score to a fact based on the distance between subject and object from each triplet in prediction.
- Category Filter: For all the annotated Wikipedia concepts in the prediction and facts, we build cat-

egory vectors and assigned a score based on the cosine similarity between prediction category vector and the fact category vector.

- Wikipedia concept relevance: Cumulative pairwise similarity score of extracted Wikipedia concepts from the prediction and fact’s context from the Wiki article.
- Context similarity: Distributional semantic similarity score between words and phrases from the prediction and fact.

From these candidate facts, we filtered the top 100 facts sorted with their current score.

## 4.3 Validation of Predictions

Our approach allows to translate the prediction and fact to a semantic representation, incorporating knowledge from external sources and then try to determine if the representation of the prediction is subsumed by that of the fact.

We pass all the (prediction, fact) pairs to two components: 1. (RATSR) framework(described below) and 2. an RTE system which performs rich syntactic analysis of the linguistic phenomena between the entailment pair.

**Relation Alignment for Textual Similarity Recognition (RATSR)** The RATSR framework has three major components: 1. Preprocessor. Prediction and fact pairs are annotated with a range of analytical tools. 2. Graph Generator. Applies metrics to compare triplets in specified annotation views to generate a match graph over the Prediction and Fact constituents of the entailment pair. 3. Alignment Score. Filters the edges in the match graph to focus on a scoring function based on the alignment output.

(1)Preprocessor: Sentence and word segmentation; POS tagging; dependency parsing; named entity recognition; co-reference resolution; temporal expression identifiers; Wikipedia concepts annotator; Multi word expression identifiers<sup>4</sup>; Phrasal verbs identifiers; Quantifier and Qualifier references. These resources are used for annotating both predictions and facts at the sentence level and triplet level.

(2)Graph Generator: Similarity metrics are applied to the relevant constituent pairs drawn from the Prediction and Fact. [2] uses relation triplet similarity by calculating similarity across subject, verb and object pairs from PPDB[12], as a feature for stance classification. We construct a relation match graph(RMG) by iterating over each triplet in prediction and fact and calculate similarity over various views to give a similarity score between the two triplets being compared

<sup>2</sup><https://www.mediawiki.org/wiki/API:Search>

<sup>3</sup><https://tagme.d4science.org/tagme/>

<sup>4</sup><https://radimrehurek.com/gensim/models/phrases.html\#id2>

and create an edge with similarity score as the weight. We propose methods for similarity between triplets for various annotations mentioned in the pre-processing step.

- Triplet Similarity Score using Latent Semantic Analysis Models (Score =  $S_1$ ): Adapting the implementation from [9] and using multiplication as vector composition operator for phrases with more than one word, we define the similarity of SPO triplets using distributional models as given below:

Probability that fact triplet  $t_f:(s_f,v_f,o_f)$  implies prediction triplet  $t_p:(s_p,v_p,o_p)$  is

$$P(t_p - > t_f) = P(s_p|t_f)(1 - P(s_p)) + P(v_p|t_f)(1 - P(v_p)) + P(o_p|t_f)(1 - P(o_p)) \quad (1)$$

$$P(s_p|t_f) = P(s_p|s_f) + P(s_p|v_f) + P(s_p|o_f) \quad (2)$$

- Triplet Similarity Score using Lexical Semantic Models (Score =  $S_2$ ): We calculate similarity scores between subject, predicate and object pairs from prediction and fact from synonym and antonym similarity using Wordnet, PPDB and Wikipedia concept Similarity; hyponym and hypernym similarity using Wikipedia and Wordnet taxonomy structure; length of the path between two entities in DBpedia; Numeric references similarity. We then combine these scores to give a cumulative lexical similarity score between the two triplets.

(3) Alignment: The goal of alignment component is to decompose the text and hypothesis into semantic constituents, and determine which prediction triplet should be aligned to which fact triplet. In contrast to aligning words[2] from prediction to fact, we align triplets to exploit the semantic roles of the constituents; to facilitate for the analysis of specific prediction attributes (in Table 1) which are matched in the fact; and also to validate against a cluster of relevant facts. We used a maximum weight perfect bipartite graph matching algorithm to align triplets from prediction to relevant triplets from facts.

From the similarity scores obtained from the RATSR framework and an RTE system[?], we set threshold limits to label the entailment pair as true, false or unrelated. .

## 5 Results & Discussion

**Dataset Preparation:** We collected two datasets, one from predictions in sports domain and the other from campaign promises made by Barack Obama. We automatically extracted predictions from articles on

Rio Olympics from 6 sites (denoted as A<sup>5</sup>, B<sup>6</sup>, C<sup>7</sup>, D<sup>8</sup>, E<sup>9</sup>, F<sup>10</sup>) and manually filtered the predictions which can be objectively evaluated and those which can be reduced to factoid questions. ‘Olympics Predictions’ dataset consists of 97 predictions made for various events in trials for Rio Olympics and the Rio Olympics 2016. We further manually annotated each prediction as true, if it has come *true* and *false* otherwise. We collected the second dataset ‘Obama Promises’ from politifact<sup>11</sup>, where each promise is labeled as ‘broken’ or ‘promise kept’ or ‘compromised’. We collected a set of 257 such promises which can be objectively evaluated.

We evaluated the predictions and obtained labels using our prediction validation system on the two datasets. Table 3 compares the accuracy scores of these labels against the actual labels. Table 2 presents the reliability scores obtained (normalized by the number of predictions) of the 6 sources we considered.

Table 1: Credibility scores for the news sites

Source	True Predictions	False Predictions	Factual Statements	Credibility Score
A	14	7	67%	29.3
B	4	9	88%	17.4
C	7	6	72%	31.1
D	7	4	69%	37.2
E	7	10	71%	21.3
F	9	13	66%	21.7

Table 2: Results for predictions validation

Dataset	TP	TN	FP	FN	Fscore
Rio Olympics	37	29	20	9	0.718
Obama Promises	111	30	26	93	0.651

**Discussion:** ‘Obama Promises’ contains multi-sentence predictions and requires more robust NLP modules to identify the main predictive clause that has to be validated, besides other supporting predictive clauses (example: “**Create a \$10 billion fund to help homeowners refinance or sell their homes.** The Fund will not help speculators, people who bought vacation homes or people who falsely represented their incomes”). High false negative rate can be attributed

<sup>5</sup><https://www.eurosport.co.uk>

<sup>6</sup><http://edition.cnn.com/>

<sup>7</sup><https://www.foxsports.com.au>

<sup>8</sup><http://www.couriermail.com.au/>

<sup>9</sup><https://www.theguardian.com/>

<sup>10</sup><https://www.thehindu.com/news>

<sup>11</sup><http://www.politifact.com/truth-o-meter/promises/obameter/browse/>

to the drift in both facts retrieval module and validation module, due to other insignificant predictive clauses. ‘Rio Predictions’ contains mostly event-based predictions and the high false positive rate for this dataset is partly due to omitting explicit negative entity similarity in the context of a given prediction. For example, the entities ‘Ussain Bolt’ and ‘Wayde van Niekerk’ are negatively related in the context of ‘winning a medal at Rio Olympics’. This negative similarity should be translated to negative triplet similarity and further to labeling as a contradicting relation for the prediction-fact entailment pair. We plan to address this in our future work, by generating alternative statements for a prediction by automatically identifying the doubt unit in a sentence and filling with relevant comparable entities/phrases.

## References

- [1] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 344–354, 2015.
- [2] William Ferreira and Andreas Vlachos. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1168, 2016.
- [3] Adam Jatowt, Kensuke Kanazawa, Satoshi Oyama, and Katsumi Tanaka. Supporting analysis of future-related information in news archives and the web. In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries*, pages 115–124. ACM, 2009.
- [4] Hiroshi Kanayama, Yusuke Miyao, and John Prager. Answering yes/no questions via question inversion. *Proceedings of COLING 2012*, pages 1377–1392, 2012.
- [5] Kensuke Kanazawa, Adam Jatowt, and Katsumi Tanaka. Improving retrieval of future-related information in text collections. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 278–283. IEEE Computer Society, 2011.
- [6] Nattiya Kanhabua, Roi Blanco, and Michael Matthews. Ranking related news predictions. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 755–764. ACM, 2011.
- [7] Hideki Kawai, Adam Jatowt, Katsumi Tanaka, Kazuo Kunieda, and Keiji Yamada. Chronoseeker: Search engine for future and past events. In *Proceedings of the 4th International Conference on Ubiquitous Information Management and Communication*, page 25. ACM, 2010.
- [8] Mio Kobayashi, Ai Ishii, Chikara Hoshino, Hiroshi Miyashita, and Takuya Matsuzaki. Automated historical fact-checking by passage retrieval, word statistics, and virtual question-answering. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 967–975, 2017.
- [9] Dmitrijs Milajevs, Mehrnoosh Sadrzadeh, and Thomas Roelleke. Ir meets nlp: On the semantic similarity between subject-verb-object phrases. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, pages 231–240. ACM, 2015.
- [10] Piero Molino, Pierpaolo Basile, Annalina Caputo, Pasquale Lops, and Giovanni Semeraro. Exploiting distributional semantic models in question answering. In *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference on*, pages 146–153. IEEE, 2012.
- [11] Christina Niklaus, Bernhard Bermeitinger, Siegfried Handschuh, and André Freitas. A sentence simplification system for improving relation extraction. *arXiv preprint arXiv:1703.09013*, 2017.
- [12] Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 425–430, 2015.
- [13] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. <http://is.muni.cz/publication/884893/en>.
- [14] Yusuke Yamamoto and Katsumi Tanaka. Finding comparative facts and aspects for judging the credibility of uncertain facts. *Web Information Systems Engineering-WISE 2009*, pages 291–305, 2009.
- [15] Navya Yarrabelly and Kamalakar Karlapalem. Extracting predictive statements with their scope from news articles. In *The 12th International AAAI Conference on Web and Social Media (ICWSM-18)*, Submitted for publication.