Fact or Fiction

CHARLES LOVERING, Worcester Polytechnic Institute, USA
ANQI LU, Worcester Polytechnic Institute, USA
CUONG NGUYEN, Worcester Polytechnic Institute, USA
HUYEN NGUYEN, Worcester Polytechnic Institute, USA
DAVID HURLEY, Microsoft, USA
EMMANUEL AGU, Worcester Polytechnic Institute, USA

Subjective reporting polarizes competing viewpoints. However, helping readers to recognize subjective content leads to more impartial discussions. Towards this end, we develop machine learning models that classify sentence objectivity. We contribute a set of linguistic rules for determining sentence objectivity collated from previous work. We also develop a labeled dataset with over 5000 sentences retrieved from various news sources. Further, we evaluate traditional machine learning classification models and artificial neural networks on our dataset. The best performing model, a convolutional neural network, achieved an accuracy of 85% and an AUC of 0.933. Using our subjective-objective sentence classification model, we implement Fact-or-Fiction, an end-to-end web system that highlights objective sentences in user text. Fact-or-Fiction provides additional information, such as links to related web pages and related previous submissions.

CCS Concepts: • **Human-centered computing → Collaborative and social computing systems and tools**; **Social tagging systems**;

Additional Key Words and Phrases: Web system; social collaboration; deep learning; text classification; subjectivity classification; sentiment analysis; natural language processing.

ACM Reference Format:
Charles Lovering, Anqi Lu, Cuong Nguyen, Huyen Nguyen, David Hurley, and Emmanuel Agu. 2018. Fact or Fiction. *Proceedings ACM Hum.-Comput. Interact.* 2, CSCW, Article 111 (November 2018), 15 pages. https://doi.org/10.1145/3274380

1 INTRODUCTION

This paper investigates automated methods that distinguish objective statements (facts) from subjective ones (fiction). We do not attempt to determine if a statement is accurate, only that is objective. For example, 'The United States reached the FIFA semi-finals in 2018' is an objective, albeit false, sentence. Conversely, 'Football is a fun sport to watch' is subjective but arguably accurate. Subjective statements can be based on fact, but are emotional or opinionated. Whereas, objective sentences present material considered factual by the speaker [27].

Authors' addresses: Charles Lovering, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, MA, 01609, USA, cjlovering@wpi.edu; Anqi Lu, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, MA, 01609, USA, alu@wpi.edu; Cuong Nguyen, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, MA, 01609, USA, ctnguyendinh@wpi.edu; Huyen Nguyen, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, MA, 01609, USA, hbnguyen@wpi.edu; David Hurley, Microsoft, 1 Memorial Drive, Cambridge, MA, 02142, USA, davehur@microsoft.com; Emmanuel Agu, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, MA, 01609, USA, agu@wpi.edu.

2573-0142/2018/11-ART111 $15.00
https://doi.org/10.1145/3274380

[19] found that modern society's increased polarity is marked by a rise in the coupling of ideology and identity. Self-identification as a partisan, one who has particular political leaning rather than a set of personal beliefs, leads to hostility - "affective polarization" [2, 6, 19]. According to [15], inter-party hostility can lead partisans to distrust the government when the other side is in control. Opinionated news has also been a catalyst for the polarization; partisans perceive less bias in opinionated news than in non-opinionated news when they are aligned with that bias [10].

Therefore, automated methods for reliably differentiating between objective and subjective statements, and presenting this information could help readers become cognizant of polarized material. Highlighting objective statements may help readers discern what is factual. Thus, we designed machine learning models to classify sentences as subjective or objective. Text classification is an active research area [3, 17]. There is prior work that classifies sentences as objective or subjective [30], and extracts opinions from texts [28]. This is one of the many sub-domains of Sentiment Analysis [10, 21], which includes a range of tasks including objectivity classification and opinion summarization.

We also provide a web interface that highlights objective sentences in text, the first step of the fact-checking process, which impacts multiple domains. Politically, isolating objective sentences can provide a common ground for people with different ideologies. This then may foster conversations based on facts rather than subjective content. Educationally, it could also help students develop their critical thinking and writing skills. Lastly, the interface provides links to articles related to the sentence in question, so readers can easily inform themselves.

Contributions

Our goal is to encourage constructive discussion by creating a platform that highlights objective sentences from text entries using machine learning. In accomplishing our goal, we made the following contributions:

(1) Collated a set of linguistic rules for determining sentence objectivity.
(2) Developed a labeled dataset of 5000 sentences retrieved from various news sources.
(3) Evaluated baseline machine learning performances on our new dataset.
(4) Deployed a user-friendly web platform that utilizes our machine learning model.

2 RELATED WORK

We discuss related work that classifies objectivity and subjectivity at both the document-level and the sentence-level representation. By reviewing proposed methods in a chronological order, two shifts in approaches to objectivity classification surface. First, there was a shift from classifiers that rely on hard-coded and hand-crafted features to classifiers that can autonomously learn the representation of textual data. Second, the choice of classifiers have changed from generative approach (classifiers that model probability distributions of each class; e.g. Naive Bayes) to a discriminative approach (classifiers that find boundaries between classes; e.g. logistic regression, neural networks). These shifts resulted in a significant improvement in performance.

Yu et al. [30] introduced methods to discriminate facts from opinions at both document and sentence levels. In order to classify documents, a Naive Bayes classifier was trained on a dataset of 4000 Wall Street Journal articles labeled as a) Editorial, b) Letter to editor, or c) Business, and d) News. Articles of types Editorial and Letter to editor were mapped to opinions, while articles of type Business and News were mapped to facts. Based on class occurrences in the training data, the Naive Bayes classifier assigns test data to its mostly likely class that is most likely to be correct. Evaluation of 4000 other Wall Street Journal showed that this technique achieved 97% F1-score (the harmonic mean of precision and recall).

OpinionFinder [28] is a system that identifies opinions, sentiments, and speculations in text. It used a Naive Bayes classifier, achieving an accuracy of 76% and a rule-based classifier reported to have about 91.7% precision and 30.9% recall (F1-score = 46%). The goal of this system is similar to our system because it uses a machine learning model to classify subjective and objective sentences.

The Passive-Aggressive [26] fact/opinion sentence classification algorithm trained on unigram, bigram, and trigram features. The authors trained multiple classifiers in such a way that a classifier that came before were used to reduce noise from input data, and the cleaner data was then used to train the subsequence classifier. After training on a set of sentences from 70,000 fact-based articles and 70,000 opinion-based articles, they reported the average F1-score of the cross-validation to be 85%.

The initial work in the field by [26, 28, 30] also used Naive Bayes classifiers. However, these works oversimplified the data collection by assuming that sentences have the same characteristics (subjective and objective) as their source articles. However, an article may contain sentences of both types. In general, their methods were not able to identify nuances in text. Although [26] selected a more effective classifier, it also suffered from the lack of a reliable dataset.

Classification of objectivity is in the domain of Sentiment Analysis (SA) [10]. SA includes a range of other tasks like positivity classification. Recently, there have been works interested in detecting which aspects [24] of the text contribute to the sentiment. However, we focus on sentence level classification, leaving aspect level analysis to future work.

Kim et. al [17] trained a subjective/objective sentence classifier with a Convolutional Neural Network (CNN) on word vectors, which were pre-trained on 100 billion words of Google News using an unsupervised model [12]. CNN's architecture is composed of a single convolutional layer, one pooling layer, and one fully connected layer. Despite its simplicity, the CNN achieved an accuracy of 93.4%. It is worth noting that when evaluating a variant of their model with no fine-tuning for the word embeddings, the neural network achieved 93.0% accuracy. We adopt this approach.

3 METHODOLOGY

In this section, we explain our methodology for building a system for sentence classification. Figure 1 illustrates our methodology. For shallow models, we extracted features from the text and then transformed the data before feeding them into the models.

3.1 Data Collection

We collected a dataset to train machine learning models. The format of our dataset is similar to Cornell's Movie Review Data [20] which contains sentences labeled as objective and subjective. It had 5000 objective sentences extracted from movie plot descriptions and 5000 subjective sentences extracted from movie reviews. We gathered a text corpus from news sources, as this was our targeted domain, including BBC [13], CNN [22], Fox News [22], and BuzzFeed [8][1]. Next, we manually labeled a random sample of 5000 sentences. This sample of news article sentences had 2200 objective sentences and 2799 subjective sentences. To create a consistent dataset we relied on prior linguistic literature to distinguish objective statements. Thus, we collated a set of rules for labeling the data. These rules are in Section 3.2.

As indicated above, there is a disparity between the number of subjective and objective sentences in our dataset. To balance the dataset, we supplemented our dataset by using Wikicorpus [25], which is a collection of articles from Wikipedia, and randomly sampled 609 objective sentences (and 37 subjective sentences) to balance the dataset. We labeled sentences using the same rules as

[1]These articles are public and fair use [8, 13, 22].

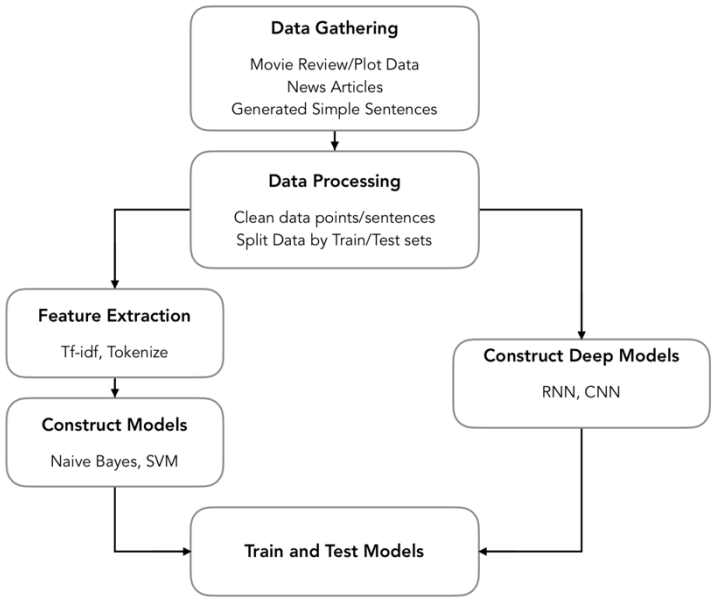

Fig. 1. Model training pipeline.

described above. Examples are shown in Table 1. Figure 2 is a histogram of how many sentences had various word counts in our corpus (news articles only).

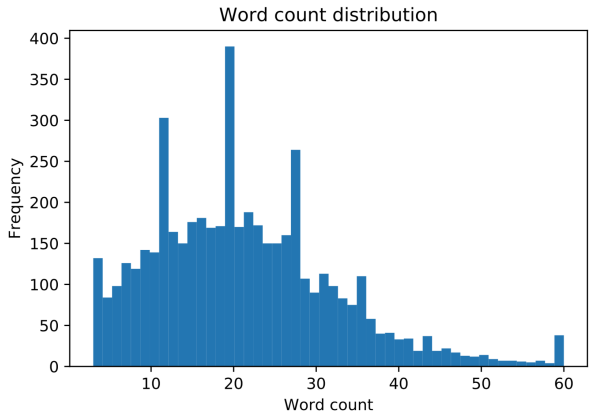

Fig. 2. Overview of the news article sentences dataset.

In order to better process informal writing style, we generated 470 objective sentences and 426 subjective sentences by substituting appropriate words into templates. Table 2 shows examples of these sentences.

In total, we had 7326 objective sentences, and 7764 subjective sentences, yielding a total of 15090 data points. We perform evaluations on both the aggregated dataset and the dataset of sentences only from news articles. The dataset statistics are detailed in Table 3.

Table 1. Example sentences from the Wikicorpus

Objective sentences	**Subjective sentences**
- Ethnic studies is an academic discipline dedicated to the study of ethnicity.	- just a granite stone with a sharp tip, almost like a spear head.
- Macke was born in Meschede, Germany.	- Probably the most significant geographical feature of Kent is the White cliffs of Dover.
- Boeing retired from the aircraft industry in 1934.	- Like many outstanding artists of her time, Bass experienced a revival of interest.
- For the past 150 years, lighting technology was mainly limited to incandescence and	

Table 2. Example sentences generated from templates.

Templates	**Sentences**
<noun>is <adjective>	- The sky is blue.
	- The Earth is round.
(Objective)	- This car is red.
	- An orange is perishable.
This <noun>is too <adjective>	- This person is too smart.
	- This dog is too good.
(Subjective)	- This job is too awful.
	- This building is too high.

Table 3. Dataset sentence count statistics.

Dataset Subset	Objective (total)	Subjective (total)	Train Size	Test Size
Aggregated	7326	7764	13411	1679
News	2200	2799	3932	492
Imdb	5000	5000	8248	1032
Generated	470	426	716	90
Wikipedia	609	37	515	65

After aggregating all sentences in a single comma-separated file, we started preprocessing each of them to conform to the following criteria:

- All punctuations are eliminated.
- All tokens are separated by a single space.
- URLs, such as www.google.com, are replaced with <URL>
- Numbers are replaced with <NUM>
- Time, such as 12:00 am, are replaced with <TIME>

We performed the last 3 substitutions to reduce the variability in our dataset and the size of the dictionary. Note that this preprocessing is also applied to data submitted to our deployed system.

3.2 Objectivity and Subjectivity Annotation Rules

We use the definition from [27]: "If the primary intention of a sentence is an ***objective*** presentation of material that is factual to the reporter, the sentence is objective. Otherwise, the sentence is ***subjective***".

Example [27]:

- At several different levels, it's a fascinating tale. *Subjective.*
- Bell Industries Inc. increased its quarterly to 10 cents from seven cents a share. *Objective.*

Private state is a general term that covers mental and emotional states, which cannot be directly observed or verified [23]. For example, we can observe evidence of someone else being happy, but we cannot directly observe their happiness. In natural language, opinions, emotions, and other private states are expressed using subjective language.

Rules:

Thus, we define a set of rules below, which determine that sentence is **subjective** if:

(1) It contains adjectives or adverbs that have an orientation (e.g. beautiful, ugly, simple, good) as opposed to adjectives that do not have an orientation (e.g. domestic, medical, red) [14].

(2) It contains explicit private state (e.g. think, believe, hope, want) or a private state mixed with speech (e.g. berate, object, praise) [29].

(3) It contains events that are not significant, speculative or not real (called "minor private states and minor speech events" in Wilson and Wiebe's study). Examples [29]:
- ***Such wishful thinking*** *risks making the US an accomplice in the destruction of human rights.* (not significant)
- *If the Europeans* ***wish*** *to influence Israel in the political arena...* (in a conditional, so not real)
- *And we are seeking* ***a declaration*** *that the British government demands that Abbasi should not face trial in a military tribunal with the death penalty.* (not real, i.e., the declaration of the demand is just being sought)
- *The official* ***did not say*** *how many prisoners were on the flight.* (not real because the saying event did not occur)
- *No one who has ever studied realist political science* ***will find this surprising.*** (not real since a specific "surprise" state is not referred to; note that the subject noun phrase is attributive rather than referential [9])

(4) If a sentence contains a quotation comprised of multiple sentences, classify the sentences in the quotation separately.
- *"Today is a good day. The sky is blue", said Bill Gates.* (First **subjective**, second **objective**).

(5) For sentences that contains nested source (A said that B thought that C did something was bad) [29], we consider the writer's point of view. For example:
- *The South African Broadcasting Corp. said the song "Freedom Now" was "undesirable for broadcasting."* According to [27]: "there is no uncertainty or evaluation expressed toward the speaking event. Thus, from one point of view, one might have considered this sentence to be objective. However, the object of the sentence is not presented as material that is factual to the reporter, so the sentence is classified as subjective (or subjective speech-event sentence to be exact)".
- *Bill Gates said he recently started using an Android smart phone.* By the same logic, this sentence is objective.

(6) If a sentence contains modal verbs: can, must, should, etc:
- *You should not leave the light on when you go to sleep.*

(7) Conjured causation relationships:
- *Thanks to Donald Trump, 200 Marines were able to see their families that day.*

All sentences which are not subjective, are objective if they are also NOT:

- Incomplete sentences. E.g. *Hey!*
- Questions. E.g. *Who are you?*
- Imperative sentences. E.g. *Go to sleep.*

3.3 Classifiers

We provide a baseline performance using prior work, notably Naive Bayes classifiers and CNN [18]. Besides Naive Bayes and CNN, we also experimented with Support Vector Machines (SVMs) and Long Term Short Memory (LSTM) [16].

We partitioned the dataset into train and test sets. To avoid data snooping, we further divided the training dataset into a validation set which we used to inform our model design. Although we were generally interested in achieving high classification accuracy, we also measured the recall, precision, and F1-score to ensure that the model was generalizable. After validation, we evaluated our models on the test sets.

3.3.1 Feature Extraction and Model Training. As our training data is in the form of text, we extract numerical features for the SVM and Naive Bayes models. We used Natural Language Toolkit (NLTK), an open source Python package [1] to help us extract features. Our features included:

- *Tf-idf of unigram, bigram, and trigram.* Term Frequency-Inverse Document Frequency, or tf-idf, measures the relative importance of different words that appear in the document by normalizing the term frequency (tf) with the inverse document frequency (idf); this weights common words as less important and rare words as more important [11]. Transforming text data into this feature is a popular method for vectorizing text [26, 30].
- *Part of speech (POS) tag count.* For example, "heat water in a large vessel" would be tagged as "<verb> <noun> <preposition> <determiner> <adjective> <noun>". Wiebe et. al [27] pointed out that the presence or absence of a particular part of speech can be a valuable signal for determining subjectivity. We used "pos_tag" module in NLTK package to obtain this feature.
- *Word sentiment score.* This feature was included in the classifiers in [30]. Each word is assigned with one of the three sentiment scores: positivity, negativity, and objectivity. We experimented with such assignments from the SentiWordNet lexical resource [5]. We did not use positivity and negativity score because objectivity score is calculated based on positivity and negativity score in NLTK's implementation: objective score = 1.0 - (positive score + negative score)". In short, objectivity score reflects sentiment polarity. Since we only needed to know the sentiment polarity rather than the sentiment itself existing in one sentence, we used just the objectivity score. The API takes in a word and its POS tag and outputs an objectivity score between 0 and 1. For example, "beautiful" as an adjective has an objectivity score of 0.25, whereas"red" as as an adjective has score of 1.0. Most verbs and nouns have objectivity scores close to 1. We combined the POS tags in a sentence to obtain objectivity score of each word.

The result of the feature extraction process is a sparse matrix, with rows that correspond to sentences and columns that correspond to features, e.g. a tf-idf value of a unigram, count of a part of speech. In the Naive Bayes classifier, we used the multinomial distribution to model our data. For the SVM, we validated with both linear and radial basis function (RBF) kernels. For deep learning models, we did not extract raw features. Each word in a sentence is transformed into a one-hot vector, a vector $v \in R^V$ where V is size of the dictionary and v a value of 1 at the location corresponding to the word it represents and 0 everywhere else. This sparse one-hot vector is then linearly mapped by embedding layer $E \in R^{V \times W}$ to a dense vector $u \in R^W$. We experimented with both training the embedding layer from our data and using pre-trained word embeddings [12]. In

our experiments, we used $W = 300$ without validation, and V is the number of unique words our dataset after preprocessing.

Our LSTM model consists of a recurrence layer and a dense layer. To accelerate the training process, we used the ADADELTA [31] to optimize cross entropy loss. We validated across a range of different sizes for the layers and found that relatively few layers worked best given the size of our dataset. For the best LSTM result reported below, we used a size of 100 for both recurrence layer and the dense layer.

We adopted the CNN architecture from [17][2]. It has an embedding layer followed by convolution, a max pool, and dropout layers. For non-linear activation function, we used the Rectified Linear Unit (ReLU) [18]. ReLU is a simple max function that activates at zero: $\text{ReLU}(x) = \max(0, x)$.

3.4 Classification Results

Table 4 shows the results of the two machine learning (Naive Bayes and SVM) and two deep learning (LSTM and CNN) models, which were evaluated using 3 different metrics: AUC, accuracy, and F1-score.

Table 4. Classification results.

Model	AUC	Accuracy	F1-score
LSTM	0.9023	0.8289	0.8162
CNN	**0.9330**	**0.8552**	**0.8552**
Naive Bayes	0.8409	0.8418	0.8328
SVM	0.8468	0.8466	0.8434

LSTMs are able to learn long-term dependencies in sequences. Conversely, CNNs is effective on capturing spatial features and operates on segments of a sentence. Similarly, Naive Bayes and SVM relied on discrete feature values of signals such as the phrase "I think" or "In my opinion" to classify sentences. The SVM, Naive Bayes, CNN models predominately outperformed the LSTM. This might be explained by the fact that when humans determine whether a sentence is subjective or objective, signal words or specific phrases are often more important than the meaning of the whole sentence. We provide further statistics for the CNN, in Table 5, detailing how well it works on various subsets of our dataset. The CNN performs well on all the subsets. It is reassuring to see that it is able to completely capture the generated data. Notably, the AUC for the wikipedia dataset is low. However, this is explained by the fact that it is predominately objective and strongly penalized for incorrect classifications by AUC.

We show how effective training the datasets jointly is in Table 5. The difference in performance from training on only the respective dataset is denoted as δ in the table. In all cases the performance difference is slight decrease in performance or identical. It is important to note that the training and testing dataset that comprise the aggregated dataset consists only of the data points from the train and test partitions of its subsets. While these results do not show that adding annotated information is useful when the domain of the subject is different, it does demonstrate that the same model is able to capture the various distributions at approximately the same performance level.

We used grid search to look for the best parameters when tuning hyperparameters for Naive Bayes and SVM models. For both models, we tried three sets of Part of Speech tagging (POS): (1) All types of adjectives, all types of verb, all types of adverb, and all types of nouns, (2) Just all types

[2]We adapt the implementation of this work provided here: https://github.com/dennybritz/cnn-text-classification-tf.

Table 5. Further CNN classification results.

Dataset Subset	AUC	Accuracy	F1-score
Trained on Respective Subset			
Aggregated	0.9330	0.8552	0.8552
News	0.8641	0.7987	0.7990
IMDB	0.9647	0.9031	0.9031
Generated	0.9980	0.9777	0.9778
Wikipedia	0.5911	0.8923	0.8415
Trained on All Data	AUC (δ)	Accuracy (δ)	F1-score (δ)
Aggregated	0.9330	0.8552	0.8552
News	0.8273 (-0.0368)	0.7500 (-0.0487)	0.7513 (-0.0477)
IMDB	0.9624 (-0.0023)	0.8972 (-0.0059)	0.8972 (-0.0059)
Generated	0.9980 (+0.0000)	0.9666 (-0.0110)	0.9667 (-0.0111)
Wikipedia	0.6453 (+0.0542)	0.8307 (-0.0615)	0.8250 (-0.0165)

of adjective, and (3) None. The results show that all three sets of POS tagging have the mean test scores of 0.8378. Therefore, the POS tagging feature was not a predictive feature in the task.

For the SVM model, we tried both linear and RBF kernels. For RBF kernels, we further tried gamma values of 0.001 and 0.0001. The linear kernel had a mean test F1-score of 0.8375 whereas RBF kernel for both gamma values only had a mean F1-score of 0.5157 for the test set. It is unclear why the RBF kernel performed poorly. We speculate that it has to do with the relatively large number of features, compared with the number of training examples. The number of features used in for both the linear and RBF kernels was directly proportional to the size of our vocabulary (100,000), and was applied to each td-idf n-gram and objectivity score, yielding a total number of features which was greater than our ≈13,000 training examples. Unlike the deep models that embed inputs into a dense vector space, shallow models work directly in this high-dimensional feature space. A simpler model, the linear kernel, trained faster than the complex RBF kernel. Reducing the vocabulary embedded for the RBF kernel would simplify the model and may allow it to generalize better.

3.5 Fact or Fiction Web Application Architecture

Figure 3 illustrates the high-level architecture of the Fact or Fiction system. We implemented our web application using the Model View Controller (MVC) pattern.

This pattern separates the whole application into three logical groups of components: models, views, and controllers. Models directly manage the application data such as user information and submitted text entries. Views are responsible for displaying data to the end users. Controllers contain the main logic of the application which might involve accessing or updating the models and changing the state of the views. By separating these components, the MVC pattern significantly promotes low coupling and high cohesion, leading to better reusability and maintainability of code.

In addition to the core application, which includes user authentication and authorization logic, HTTP endpoints logic, and database management logic, we isolated the machine learning model service (ML service), sentence embedding service and API for information related to the sentences as three external entities. The core application communicates with these entities via an API call, e.g. HTTP request/response. With this design, we were able to easily build the three entities, switch any technologies when necessary and reuse existing third-party services. The interface of these

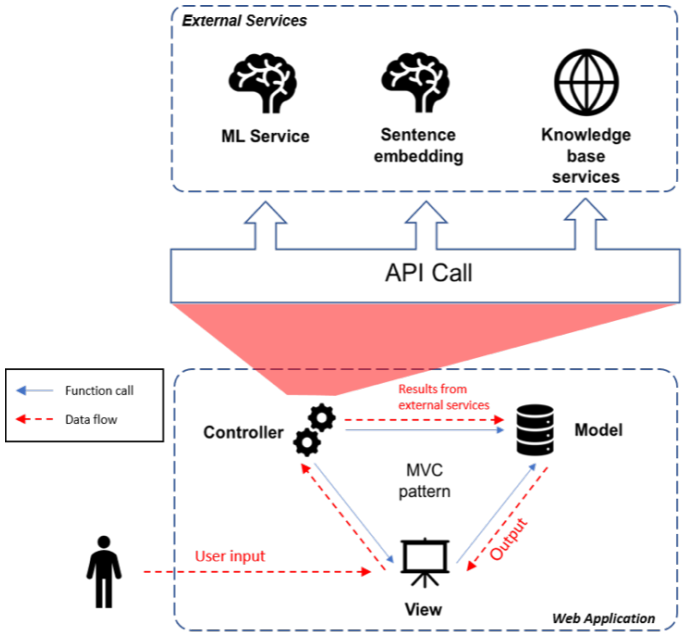

Fig. 3. Architectural design of the system

external services is rather simple: they all accept a piece of text or a list of sentences. The ML service returns a classification decision for each of the sentences, i.e. objective or subjective. The sentence embedding service outputs an embedding vector for each sentence it receives. We then used the embedding vector to find similar sentences. The knowledge base services fetch related information such as links to relevant articles.

An end-to-end data flow starts from the user entering text input into a view component. This text input is then transferred to a controller, which in turn converts the user input into appropriate formats that conform to each external service and simultaneously sends to all of them. After receiving responses from the external services, the controller transforms these results into the corresponding models, which are later displayed to the user via the view component. This flow is summarized by the red arrows in Figure 3.

4 IMPLEMENTATION

We built Fact-or-Fiction front-end using React, Redux and Microsoft Fabric UI components, and the back-end using ASP.NET Core MVC and Microsoft Azure Machine Learning. In this section, we demonstrate the implementation of Fact-or-Fiction web system that interfaces with the user. [3]

4.1 Fact or Fiction Application Required Functionality

Based on our objectives, we set the following high-level requirements for the final application.

- The application can classify objective and subjective sentences in text input in real time.
- The user can see related sentences by selecting any sentence.

[3] A clone of our repository is available at https://github.com/cjlovering/FactFiction and a short demonstration of the work is at https://youtu.be/mgnwe0zsQv0.

- The application will provide information related to a sentence in order to assist the user in fact-checking.
- The user can vote on the truthfulness of any sentence.

4.2 Fact or Fiction Application User Interface

The primary purpose of the Fact Fiction web interface is to showcase the capabilities of the sentence classification technology in a clear and complete manner. It has a single view - a form that allows a user to paste textual content. Then user receives contextual information that expands upon user interaction and interest.

We built a scalable website hosted on Azure Web services with a clean and friendly interface using React, Redux, and Office Fabric UI. The interface includes four different views: Welcome screen, Login screen, Text input and feed view (Figure 4) and Result view (Figure 5).

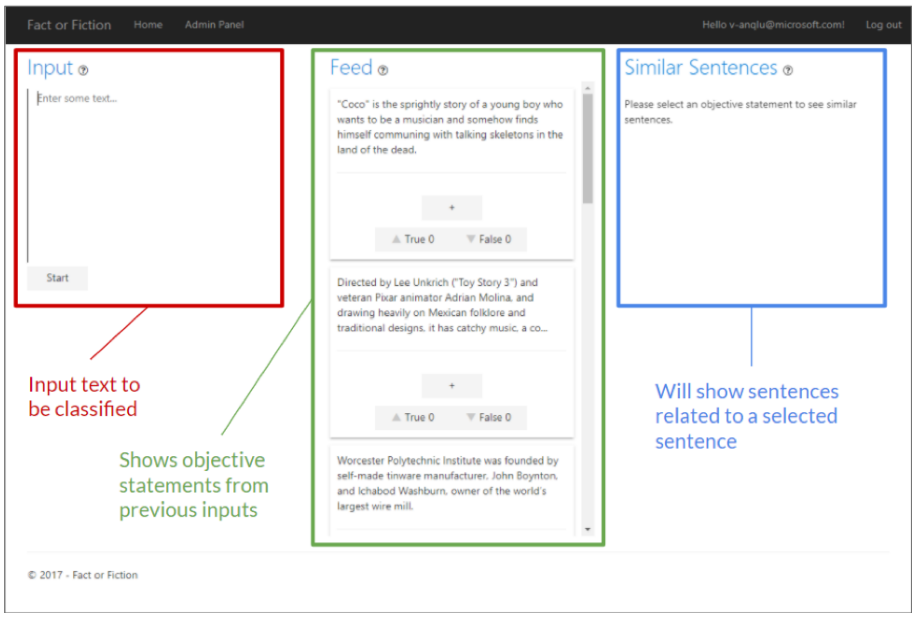

Fig. 4. Text input and feed view.

Once logged into the system, a user sees home view shown in Figure 4, which has three columns:

(1) **Input**: where users copy and paste or input a block of text they would like to analyze in the input box.
(2) **Feed**: objective sentences extracted from other users input, sorted in descending order by the time. Once a user scrolls to the bottom, the feed will load more sentences. Each sentence is presented in a card that shows votes others have cast and can show more details by clicking on the "+" button, illustrated in Figure 5.
(3) **Similar Sentences**: Once a user selects an objective sentence card, the similar sentences column will display a maximum number of five other objective sentence cards that are the most related from the previous submissions of all users. This functionality is illustrated in detail in Figure 6.

The question mark icon next to the column titles indicate tooltips that describe the purpose of each column. After the user inputs text, clicking the "Start" button will bring the user to the result view shown in Figure 5.

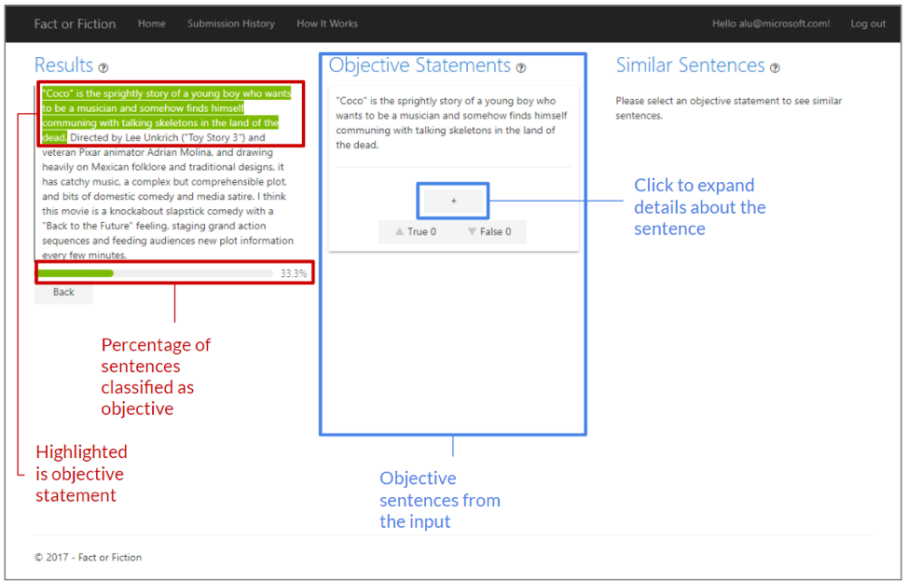

Fig. 5. The Result screen.

The Result view (shown in Figure 16) contains three columns:

(1) **Results**: This column shows the classification results by highlighting objective sentences in green. A user can select a sentence to see the corresponding objective sentence card and the card automatically scrolls into the view. The bar below the text shows the percentage of objective, input sentences.
(2) **Objective Statements**: The cards in this column (left side of Figure 6 corresponds to each objective sentence highlighted in the Results column.
(3) **Similar Sentences**: These are sentences are similar to any selected objective sentence (right side of Figure 6).

When a user clicks on a card, the card is selected and highlighted. Our server then fetches at most five similar objective sentences from our database. As shown in Figure 6, the similar sentences do not necessarily share the similar words syntactically; instead, it analyzes the intent of the sentence and returns the sentences with similar intentions. In Figure 5, the selected sentence is about the Pixar animated film, *Coco*, and the similar sentences shown are about the plot of *Toy Story* and about Pixar. We do not describe how we do this here as it is not one of our primary objectives or contributions.

Within each objective sentence card, a user can get more information about the sentence by clicking on the "+" button. Once clicked, the card expands to show a table (Figure 6). The "Related information" section provides links that are related to the sentence. The "Recognized entities" section shows entities that are extracted from the sentence. If available, "Site bias" shows potential bias from links provided. The user can also click on the "-" button to hide this information.

Since our model only classifies whether a sentence is objective or not, it does not tell if a sentence is actually true or false. It is up to users as a community to fact-check the sentences. By using

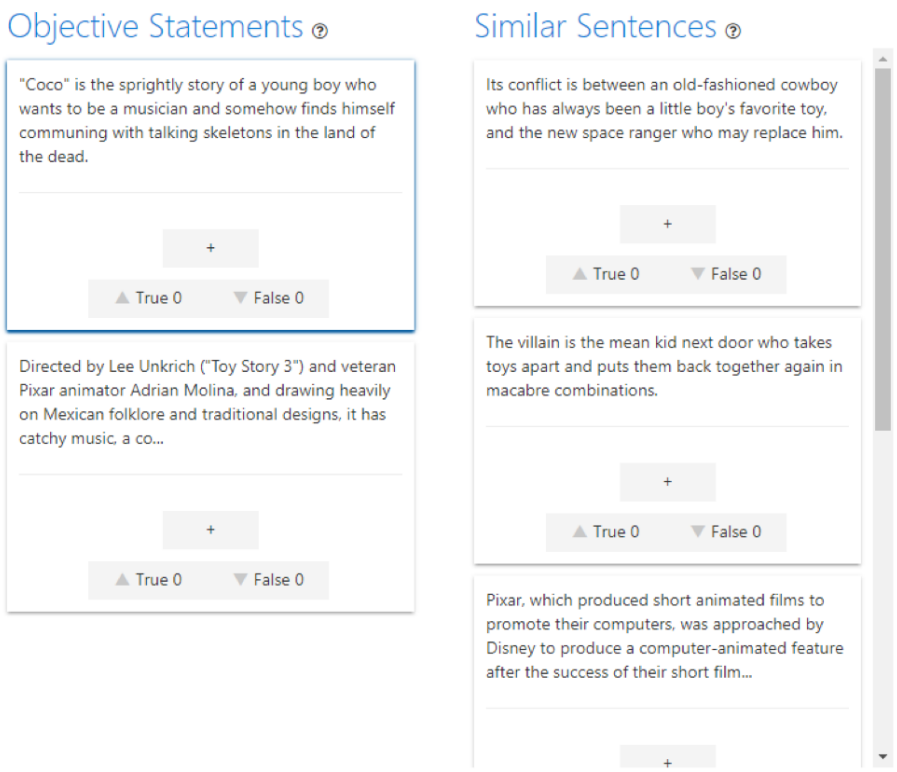

Fig. 6. The Similar Sentences feature.

their prior knowledge and the references link provided by Fact or Fiction, users can determine the veracity of sentences. After that, they can vote sentences to be true or false through the buttons. One user can only vote once on the same sentence. Canceling and changing votes is supported.

We conducted anonymous surveys and user interviews to evaluate the user experience of our platform. In general, the users found the website useful and credible.

5 CONCLUSION AND FUTURE WORK

5.1 Conclusion

We contributed a web platform to foster constructive, objective discussion based on data. To do this, we built a high-accuracy model that extracts objective sentences from a document. When a user inputs text, the platform extracts and highlights the objective sentences and presents them directly to the user. This allows users to fact-check statements with external resources. We are motivated to help reduce the negative effects of opinionated news [7, 10]. Helping readers become actively aware of opinions in documents they read and focus on the facts, may help ameliorate this issue. The performance of CNNs on our dataset is $\approx$ %85. We found that the shallow models, SVMs and Naive Bayes, had similar performance, whereas the recurrent neural networks model did not perform as well (see 3.4). We postulate that more data is necessary to fully utilize this more complex model.

5.2 Future Work

Generalization - With a crowdsourcing tool like Mechanical Turk [4], a qualifier test can be administered to find annotators for those who understand the objectivity rules (Section 3.2). Next, a much larger dataset can be collected in an extensible and affordable manner.
Internationalization - A limitation for our work is that it only works for English; we envision a system that functions across languages. This is difficult as our machine learning model requires training data for any given language, but we postulate that machine translation could help solve this problem.
Accessibility - Integrating this pipeline directly into the media habits of readers will allow them to read in a more informed manner. A concrete example of this would be creating a tab extension using our service.

ACKNOWLEDGMENTS

This work was funded by Microsoft's Garage program as a Worcester Polytechnic Institute (WPI) Major Qualifying Project (MQP). This program was organized by Ben Fersenheim.

REFERENCES

[1] [n. d.]. *Natural Language Toolkit.* https://www.nltk.org/.
[2] Alan Abramowitz and Kyle Saunders. 1998. Ideological realignment in the US electorate. *The Journal of Politics* 60, 3 (1998), 634–652.
[3] Charu Aggarwal and ChengXiang Zhai. 2012. A survey of text classification algorithms. In *Mining text data*. Springer, 163–222.
[4] Amazon. [n. d.]. *Amazon Mechanical Turk.* https://www.mturk.com/mturk/welcome.
[5] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining.. In *LREC*, Vol. 10. 2200–2204.
[6] Larry Bartels. 2000. Partisanship and voting behavior, 1952-1996. *American Journal of Political Science* (2000), 35–50.
[7] Mark Boukes, Hajo Boomgaarden, Marjolein Moorman, and Claes De Vreese. 2014. News with an attitude: Assessing the mechanisms underlying the effects of opinionated news. *Mass Communication and Society* 17, 3 (2014), 354–378.
[8] BuzzFeed [n. d.]. *BuzzFeedNews Github.* BuzzFeed.
[9] Keith Donnellan. 1966. Reference and definite descriptions. *The philosophical review* (1966), 281–304.
[10] Lauren Feldman. 2011. Partisan differences in opinionated news perceptions: A test of the hostile media effect. *Political Behavior* 33, 3 (2011), 407–432.
[11] William Frakes and Ricardo Baeza-Yates. 1992. *Information retrieval: Data structures & algorithms*. Vol. 331. prentice Hall Englewood Cliffs, New Jersey.
[12] Google. 2013. *word2vec.* https://code.google.com/archive/p/word2vec/.
[13] Derek Greene and Pádraig Cunningham. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd international conference on Machine learning*. ACM, 377–384.
[14] Vasileios Hatzivassiloglou and Janyce Wiebe. 2000. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 299–305.
[15] Marc Hetherington and Thomas Rudolph. 2015. *Why Washington won't work: Polarization, political trust, and the governing crisis.* University of Chicago Press.
[16] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[17] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).
[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.
[19] Yphtach Lelkes. 2016. Mass polarization: Manifestations and measurements. *Public Opinion Quarterly* 80, S1 (2016), 392–410.
[20] Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 271.
[21] Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval* 2, 1–2 (2008), 1–135.

[22] Mingjie Qian and Chengxiang Zhai. 2014. Unsupervised feature selection for multi-view clustering on text-image web news data. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*. ACM, 1963–1966.
[23] Randolph Quirk. 1985. A comprehensive grammar of the English language. (1985).
[24] Toqir Rana and Yu-N Cheah. 2016. Aspect extraction in sentiment analysis: comparative analysis and survey. *Artificial Intelligence Review* 46, 4 (2016), 459–483.
[25] Samuel Reese, Gemma Boleda, Montse Cuadros, and German Rigau. 2010. Wikicorpus: A word-sense disambiguated multilingual wikipedia corpus. (2010).
[26] Adam Stepinski and Vibhu Mittal. 2007. A fact/opinion classifier for news articles. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 807–808.
[27] Janyce Wiebe, Rebecca Bruce, and Thomas O'Hara. 1999. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*. Association for Computational Linguistics, 246–253.
[28] Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. OpinionFinder: A system for subjectivity analysis. In *Proceedings of hlt/emnlp on interactive demonstrations*. Association for Computational Linguistics, 34–35.
[29] Theresa Wilson and Janyce Wiebe. 2003. Annotating opinions in the world press. In *Proceedings of the Fourth SIGdial Workshop of Discourse and Dialogue*.
[30] Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*. Association for Computational Linguistics, 129–136.
[31] Matthew Zeiler. 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).

Received April 2018; revised July 2018; accepted September 2018