NLP Analysis of the Indian News Media

A Comprehensive Framework for estimating Bias

A report submitted in partial fulfillment
of the requirements for

Dual Degree

in

Computer Science & Engineering

by

Deepak Bansal

2014CS50435

Kapil Kumar

2014CS50736

Under the guidance of

Prof. Aaditeshwar Seth

Department of Computer Science and Engineering,
Indian Institute of Technology Delhi.

June 2019.

Certificate

This is to certify that the report titled **NLP Analysis of the Indian News Media** being submitted by **Deepak Bansal and Kapil Kumar** for the award of **Dual Degree** in **Computer Science & Engineering** is a record of bona fide work carried out by them under my guidance and supervision at the **Department of Computer Science & Engineering**. The work presented in this report has not been submitted elsewhere either in part or full, for the award of any other degree or diploma.

Prof. Aaditeshwar Seth
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi

Abstract

We use natural language processing and machine learning algorithms to study the bias prevalent in various media sources in the content published by them. Also, we study how various influential entities differ in their viewpoints on important policies. We have developed an entire pipeline comprising of several key steps like Entity Resolution, Latent Dirichlet Allocation based Aspect Extraction and Sentiment analysis. The analysis has been performed on several policies and events of national importance. Moreover, we have built deep-learning powered stance detection classifiers for which in-house datasets have been created and annotated.

Acknowledgments

This thesis is a product of inputs of many people who have contributed time and again with their valuable ideas and efforts.

We are immensely grateful to our project guide Prof. Aaditeshwar Seth for being highly supportive to us throughout. He has given us lot of liberty to work on different interesting dimensions in the project. He has been very active and regularly provided us valuable guidance and feedback. We feel that we have learned a lot from him and we could explore many directions only because of him. We would like to sincerely thank him for constantly motivating us and pushing us to achieve higher.

Also, we would like to thank our senior Anirban Sen for attending to all our queries and mentoring us throughout the project. A special thanks to our friends Ayush Gupta, Sourabh Basutkar and Bipul Kumar for their kind help.

Deepak Bansal and Kapil Kumar

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The world is entering into an era of increasing social media and mass media discussions about various events occurring globally. In this study, we aim to explore the bias associated with different Indian news sources. The overall bias comprises coverage bias and sentiment bias. For estimating the bias, we find out the various topics and the extent to which they have been discussed in a news article. We leverage the **Latent Dirichlet Allocation (LDA)** algorithm for topic modeling over the articles and obtaining the aspects being talked about. We have talked about the LDA technique and it's applications in Chapter 2.

We have developed an entire pipeline comprising of several key steps like **Entity Resolution** , **Latent Dirichlet Allocation based Aspect Extraction** and **Sentiment analysis**. The analysis has been performed on several policies and events of national importance. Chapter 3 describes a comprehensive analysis on **'Aadhar'** which has been discussed and debated widely in the media for years. We have done the bias analysis over several datasets, **Parliament Question Hour** Dataset being one of the important ones as covered in Chapter 4

It is important to spot out the entities in a news article so as to find the By statements i.e. the statements spoken by those entities. Therefore, **Entity Resolution** being one of the most important part of our analysis, we have developed Entity Resolution algorithms and implemented several optimizations on the top of it to maintain a live entity resolution task on the continuously crawled articles. It has been exhaustively covered in Chapter 5.

Apart from news articles as a whole, we deeply analyze the statements given by various influential people and find out the stance taken by them on different policies of national importance. These people might be politicians across different parties, bureaucrats, entrepreneurs, policy experts, activists among

others. Further, these biases are aggregated across parties or organizations to determine the ideological position held by a certain entity group.

Detecting the bias in a statement is a sophisticated language processing task. Identifying the bias across statements can sometimes be simple if they have certain words that reflect a particular stance, but more often than not its not that easy. Generally, the entire semantics associated with the statement has to be captured by the model. There are therefore two approaches to the task, one is a sequential approach generally adopted by recursive neural networks or LSTMs, and the other is the hierarchical approach. In the hierarchical approach, the sentence is perceived to be built up from phrases bottom up. Recursive neural networks are one kind of model that trains on the hierarchical aspect of sentence formation.

Iyyer et al. (2014)[7] use **Recursive Neural Networks** for the task of political ideology detection. The authors show that they achieve better results than all existing methods. Recursive Neural Networks are a type of **hierarchical neural networks** which take into account both the syntactic and semantic features of the sentence. The algorithm is based on the assumption that the meaning of each phrase should be a combination of the meaning of the words in it applied recursively based on the syntax. The authors use the cross-entropy loss function for evaluation and the Limited-Memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm is used for optimizing the objective function.

Inspired from Iyyer et al. (2014)[7] we use Recursive Neural Networks to build **Stance Detection Classifiers** for our domain and also we try several modifications on the top of it to improve the results. For this task, We have developed and annotated the entire dataset in-house with the help of peers in the research group. The detailed deep learning analysis is provided in Chapter 6

Chapter 2

Latent Dirichlet Allocation

2.1 Topic Modeling

Topic Modeling is the process of discovering the different topics present in a document. Here topic (also called aspect) basically refers to the areas of discussion present in a document. In other words, topic modeling is the process of learning shorter descriptions for efficient processing of large documents while preserving their essence. Latent Dirichlet Allocation (LDA) is a very popular generative and probabilistic topic modeling scheme. It was proposed by David Blei, Andrew Ng and Michael I. Jordan in 2003 [9].

Suppose we have the following set of sentences:

- I like to eat broccoli and bananas.
- I ate a banana and spinach smoothie for breakfast.
- Chinchillas and kittens are cute.
- My sister adopted a kitten yesterday.
- Look at this cute hamster munching on a piece of broccoli.

LDA output is as follows:

- Sentences 1 and 2: 100% Topic A
- Sentences 3 and 4: 100% Topic B
- Sentence 5: 60% Topic A, 40% Topic B
- Topic A: 30% broccoli, 15% bananas, 10% breakfast, 10% munching
- Topic B: 20% chinchillas, 20% kittens, 20% cute, 15% hamster

2.2 Methodology

LDA represents documents as mixtures of topics that spit out words with certain probabilities. LDA is essentially a bag-of-words unsupervised learning model and it assumes when writing each document, we decide on the number of words N the document will have (say, according to a Poisson distribution). Then, choose a topic mixture for the document (according to a Dirichlet distribution over a fixed set of K topics). Then, generate each word w_i in the document by:

- First picking a topic (according to the multinomial distribution that we sampled above; for example, you might pick the food topic with 1/3 probability and the cute animals topic with 2/3 probability).
- Using the topic to generate the word itself (according to the topics multinomial distribution). For example, if we selected the food topic, we might generate the word broccoli with 30% probability, bananas with 15% probability, and so on.

Assuming this generative model for a collection of documents, LDA then tries to backtrack from the documents to find a set of topics that are likely to have generated the collection.

2.3 Application

We have leveraged LDA for the mass media analysis where we find out the different aspects being covered in various news sources. Also, we try to find out the aspect-wise sentiments and stance reflected in the news sources. LDA basically gives us a list of keywords for each topic and after analyzing these keywords and also by reading some of the articles that fall in that topic, we arrive at the best-possible name of that topic. The table 2.1 below aptly depicts this process over the Aadhar media corpus. The detailed LDA results for the Aadhar Media corpus can be found in the Appendix A.

There are several interesting applications of LDA as following:

- Classification and clustering
- Novelty detection
- Summarization - finding a short description of the text while preserving essential statistical relationships.
- Similarity/Relevance Judgements
- Can be used in other collections, for example, images and caption
- Can be viewed as a dimensionality reduction technique

S.No.	KeyWords (Top 50)	Topic (Manual Identification)
0	cards food ration card civil said pds department crisil pan shops supplies smart distribution price aadhar fair bpl tax holders state government income bogus shop kerosene families public number link security minister poverty linking cent consumer family supply line proposed sale pvt data lakh linked beneficiaries details district dealers india	Impact on (Public Distribution System) PDS due to Aadhar Linking
1	voters election card voter electoral identity aadhar polling id commission list number vote photo elections names said issued voting district officer assembly release details roll proof officers candidates office special members employees cards stations chief commissioner level drive april driving free photograph held constituency exercise programme polls press pan linking	Elections, Voter Identities & Aadhar
2	digital mobile said transactions payments payment india app services using service online internet use cashless technology based express platform enabled indian passengers phone used users make electronic like tickets systems download customer telecom companies access transaction aadhar 000 phones network customers banks launched pay railway latest money news business financial	Digitization, Cashless Payments (Aadhar Based Payments)

Table 2.1: LDA Analysis over Aadhar Media corpus

Chapter 3

Mass Media Analysis on Aadhar

We try to perform a comprehensive analysis on policies and events of national importance. Here, we describe the entire analysis pipeline for Aadhar which has been widely discussed and debated in the media for several years.

3.1 Data collection

- We have collected all Aadhar related articles into a collection aadhar_articles in MongoDb, 4710 Articles.
- We used the following regex to extract the articles:

  Aadhar Card— UIDAI — aadhar — aadhar card — AADHAR CARD— adhar card— adharcard —adhar —ADHAR

3.2 Entity Extraction: OpenCalais

We use OpenCalais to extract named entities of type Person, Company, Organization, City, Country, etc from an article. OpenCalais extracts entities of many types but we use only a few of them in our system. Also, only a subset of properties of an entity are being used. We use entities of type Person, company, Organization, Country, City, Continent, ProvinceOrState and properties name, type, instances (entity references in article), resolutions (unique id given by OpenCalais from its database). Extracted entities are added to the article document stored in MongoDB. Each entity is then enriched with contextual information and stored in a separate entity collection. This collection might contain duplicates. Henceforth, we will refer to this collection as unresolved_entities. OpenCalais tries to resolve extracted entities by searching them in its database, but mostly gives accurate ids for places, somewhat accurate for companies and poor for persons.

3.3 Entity Resolution : Elastic Search

Entities extracted from media articles are inserted into unresolved_entities collection. This collection contains duplicates entities and they need to be resolved. For this we create a resolved_entities (initially empty) collection which will have resolved entities. There are millions of entities in our system and comparing with each one during resolution will be time consuming. To do this efficiently, we use Elasticsearch engine. It is a highly scalable open-source full-text search engine, popularly used as an underlying engine in applications for efficiently storing and querying large amounts of data. So there are two copies of resolved_entities collection (MongoDB & Elasticsearch) in the system. To resolve an entity_x from unresolved_entities, we search for it in resolved_entities using Elasticsearch which gives ten best matched results. From these results, we find the best match for entity_x.

3.4 Topic Modelling (Aspects Extraction): LDA

We classify all the news articles into different topics which are found by LDA.

3.4.1 Parameters Tuning for LDA

- Number of features to find = 1000
- Number of topics to extract = 30
- Number of top word for each topic = 50

3.4.2 LDA Results

The table 3.1 lists the manual labelling of 19 topics identified by lda for Aadhar:

S.No.	Topic/Aspect
1	PDS & Aadhar Linking
2	Elections, Voter Identities & Aadhar
3	Digitization, Cashless Payments
4	Official Documents like Passport, Birth Certificates etc.
5	Aadhar Act, Indian Economy & Policy
6	Aadhar Enrollment
7	Measures by UIDAI
8	Crimes & Aadhar Card
9	Pensions
10	Governance, Policy & Legislation
11	Details on Aadhar Cards & Distribution of Aadhar Cards
12	Women, Minority Groups & Aadhar Registrations
13	Farmers
14	Opposition & Politics over Aadhar
15	Linking of Aadhar with other schemes
16	Court & Controversy Over Aadhar Card
17	Banks & Aadhar Cards
18	Miscellaneous
19	LPG Subsidy & DBT (Direct Benefit Transfer)

Table 3.1: 19 Manually labelled aspects for Aadhar

3.5 Sentiment Analysis (Vader/ Alchemy)

Sentiment of a text can tell us the extent to which it sounds positive/negative. We evaluated a set of sentiment extraction algorithms mentioned in the iFeel system. We prepared a gold standard of 100 news articles and 25 opinions. A label from the set Negative, Positive, Neutral was assigned independently by two volunteers. Inter-agreement percentage is 82.4. The output of each algorithm is mapped onto the same set Negative, Positive, Neutral. Sentiments extracted by AlchemyAPI was found to agree with our labels the most. But it is not an open source tool and using it for the entire media system is not a good option. We have used SentiStrength which is an open source tool and its performance is comparable with that of AlchemyAPI. SentiStrength provides a sentiment value on a scale of -4 (very negative) to 4 (very positive).

News Source	Average Sentiment
The Hindu	-0.3761408083
The Times of India	-0.5313693399
Hindustan Times	-0.7894736842
Indian Express	-0.5959885387
Deccan Herald	-0.388
Telegraph	-0.5470852018
The New Indian Express	-0.4820512821

Table 3.2: Source-wise Sentiment

Aspect	Average Sentiment
PDS & Aadhar Linking	-0.4464285714
Elections, Voter Identities & Aadhar	-0.2820512821
Digitization, Cashless Payments	-0.07936507937
Official Documents like Passport, Birth Certificates etc.	-0.2602739726
Aadhar Act, Indian Economy & Policy	-0.5165692008
Aadhar Enrollment	-0.2582159624
Measures taken by UIDAI	-0.217623498
Crimes & Aadhar Card	-1.376093294
Pensions	-0.02380952381
Governance, Policy & Legislation	-0.3461538462
Details on Aadhar Cards & Distribution of Aadhar Cards	-0.7946708464
Women, Minority Groups & Aadhar Registrations	-0.2264150943
Farmers	-0.4545454545
Opposition & Politics over Aadhar	-0.871657754
Linking of Aadhar with other schemes	-0.2015655577
Court & Controversy Over Aadhar Card	-0.819047619
Banks & Aadhar Cards	-0.07894736842
Misc.	-0.1961722488
LPG Subsidy & DBT (Direct Benefit Transfer)	-0.2147239264

Table 3.3: Topic-wise Sentiment

Table 3.2 shows the average sentiment for the seven news sources and table 3.3 depicts the topic wise sentiments.

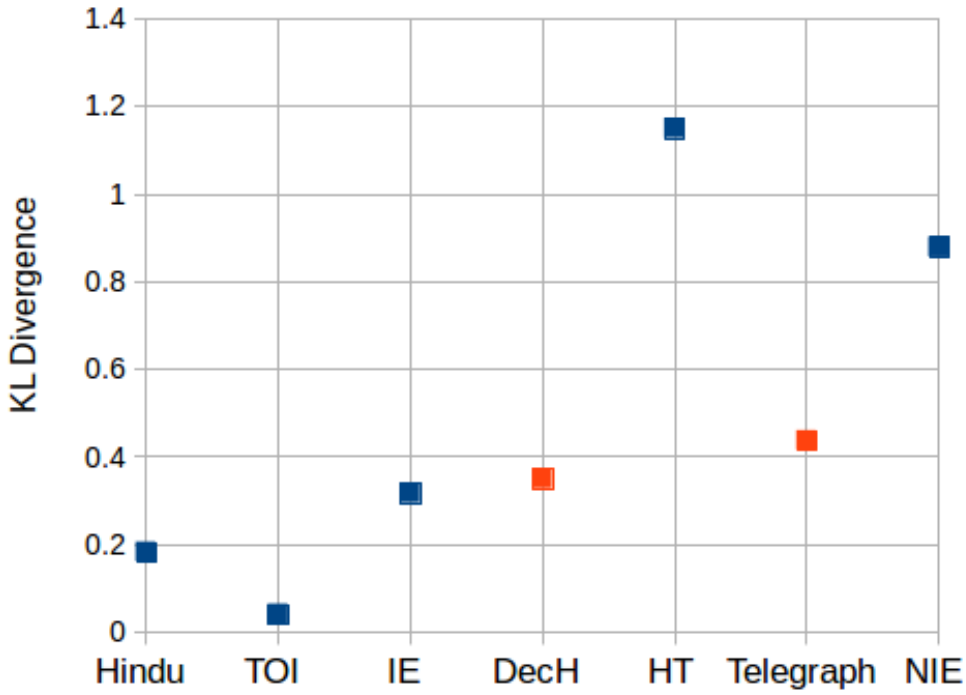


Figure 3.1: KL Divergence [News Source Coverage]

3.6 Coverage Analysis

Figure 3.1 shows the KL divergence for News Source Coverage.

Figure 3.2 shows the Relative coverage for top 20 entities of Aadhar.

3.7 Aspect-Category Mapping

Finally we mapped the aspects to one or more of the following overall categories given in table 3.4.

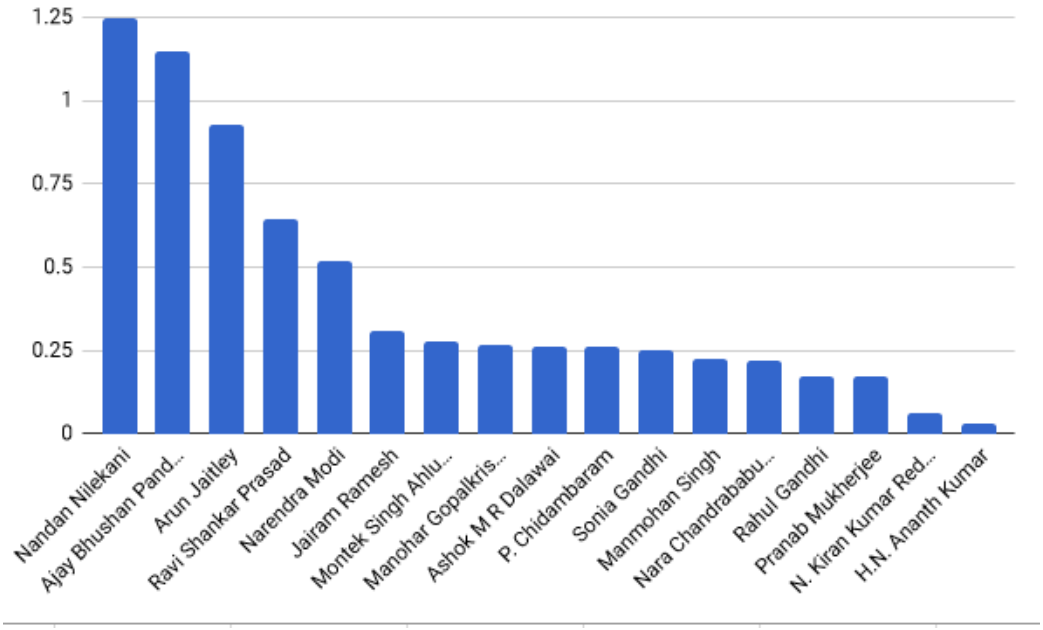


Figure 3.2: Relative entity coverage

Category	**Ideology**
Pro/Anti Welfare	to provide for the poor and help in wealth distribution
Pro/Anti Middle Class	to help the middle class (defined as those with some disposable income) grow and expand their income
Pro/Anti Neo-liberal Economics	driven by economic growth, free-market policies, minimum government, and formalization
Pro/Anti Informal Sector	driven by slow formalization of industries and trade (including agriculture)
Pro/Anti Government	whether it favours or opposes the government

Table 3.4: Constituencies

Chapter 4

Parliament Question Hour Analysis

The parliament, media, and the citizens are some of the most important stakeholders in the democratic process. While the parliament is the key stakeholder in policy formulation, mass media is an important agency through which public opinion on these policies are formed and influenced. Finally, through social media, citizens can find a voice to convey their experiences and opinions about policies, and potentially even shape the discourse even in the media and parliament. Given these duties of the key stakeholders of democracy, we have done a measurement study to understand the ways in which these stakeholders engage in the policy process, by examining the content in each of these spaces. For this purpose, we study four economic policies in India, namely, Demonetization, Aadhaar, GST, and Farmers Protest.

We analyze data on the questions asked by elected members of the Parliament (MPs) during the parliamentary question hour on the various policies between 2013 and 2017.

Our analysis also shows that the parliamentarians mostly are involved in partisanship on party lines [2], and shape their discussions based on their party goals that change over time depending on whether they are in power, or in the opposition.

4.1 Methodology

4.1.1 Collection of data

For Question Hour, we extract the data from the Lok Sabha website. We extract various metadata fields like Question ID, Question number, link, date,

Ministry, Member who asked the question, Subject of the question, and the question text. Next we filter the questions for each policy based on some manually identified keywords for different policies using regular expression matching.

4.1.2 LDA Aspect Extraction

We use Latent Dirichlet Allocation (LDA) to identify different aspects within each event, similar to [1]. LDA is a statistical model that maps a set of documents to unobserved topics, which aids in clustering similar documents into topic clusters that can be manually examined and labeled. More details about lda methodology are provided in Chapter 2

For mass media: We obtained 16 aspects for Demonetization, 14 aspects for Farmers Protest, 11 aspects for GST, and 17 aspects for Aadhaar by merging some aspects together, and then named each aspect manually. Articles are mapped to aspects if LDA gives a probability of greater than 0.3 for the mapping. We used the best performing topic coherence measure as suggested in the paper by Roder etal. [3], in conjunction with the PyLDAVis package [4], to infer the optimal number of topic clusters to be specified for each policy event. To measure the accuracy of LDA aspect mapping, two authors randomly selected 200 articles from each event and assigned aspect names for each of those articles (from the aspects labeled after performing LDA) by reading the article text and coming to an agreement. We then checked if the manually assigned and the LDA-assigned aspects for the questions match with each other. The accuracies of mapping are 85% for Demonetization, 96% for Aadhaar, 81% for GST and 76% for Farmers Protest.

4.2 Results

4.2.1 How do the statements/questions vary across the dominant political parties?

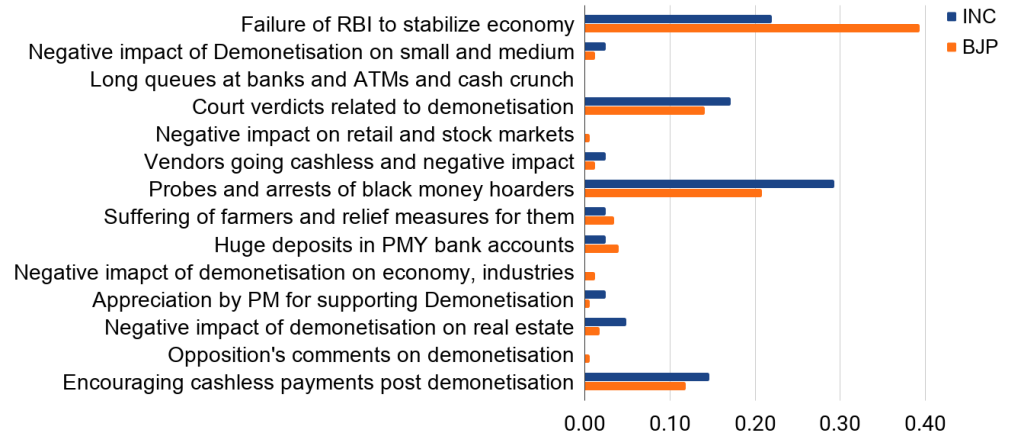
In this section, we see how the coverage of aspects corresponding to the four policies varies across the two biggest political parties in the entire QH data. We present the coverage given to each aspect for each policy by the BJP (406 MPs considering LS15 and LS16) and INC (292 MPs considering LS15 and LS16) in Figure 6.3. We find that INC tends to ask more questions in each policy, mostly around the procedural aspects and mechanics around the implementation of these policies. In the following subsections, we analyze each policy separately.

4.2.1.1 Demonetization

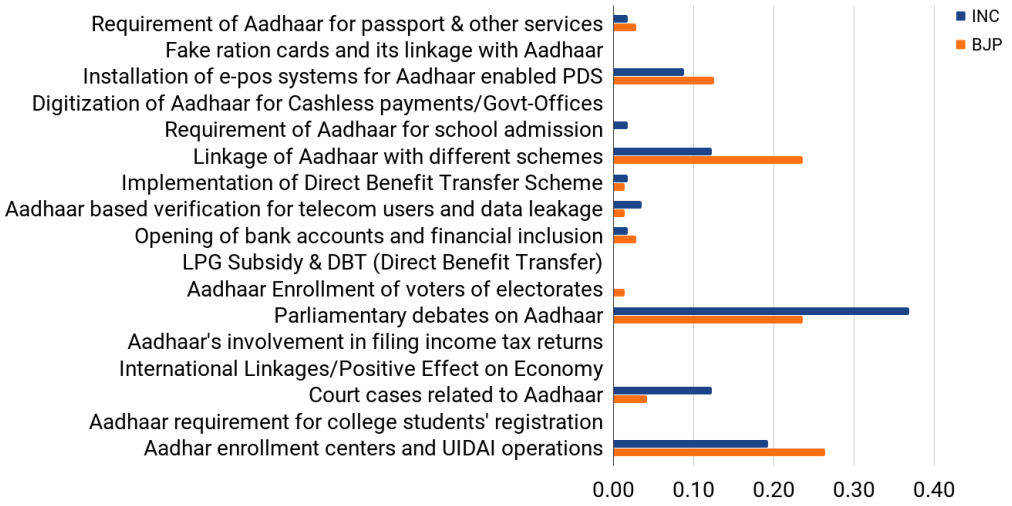
We find that a total of 205 questions on Demonetization by BJP, and 43 questions by INC. Looking at the relative coverage provided to aspects, we find that both of the parties provide nearly equal coverage for most of the aspects. The aspects [Court verdicts related to Demonetization and penalties issued to black money hoarders] (relative coverages: 0.17 by INC, and 0.12 by BJP), and [Probes and arrests of black money hoarders] (relative coverages: 0.28 by INC and 0.18 by BJP) get significantly higher coverage by INC than BJP. This is expected as being in the current opposition, INC tends to often question the governments narrative of fighting graft money and corruption. BJP provides significantly higher coverage than INC to the aspect [Failure of RBI to stabilize economy post Demonetization(relative coverages: 0.21 by INC and 0.34 by BJP)]. This aspect focuses on the procedural issues around restoration of normalcy post implementation of the policy.

4.2.1.2 Aadhaar

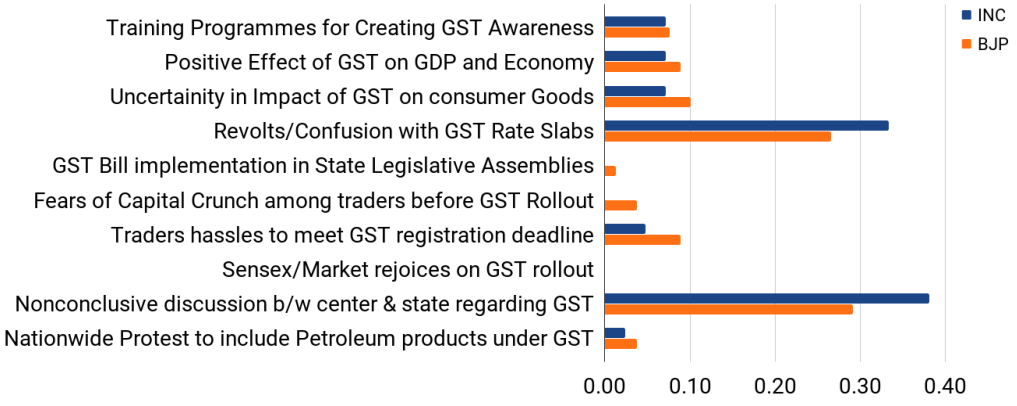
We find a total of 72 questions by the ruling party BJP, and 57 questions by the opposition party INC. We find that the opposition INC provides signifi-



(a) Demonetization

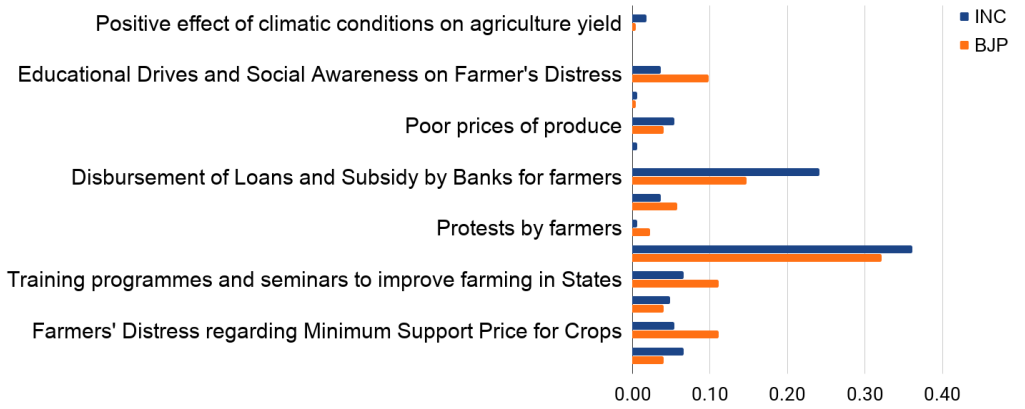


(b) Aadhar



(c) GST

Figure 4.1: Relative coverage of aspects provided by political parties in QH data for the four policies



(d) Farmers' Protest

Figure 4.1: Relative coverage of aspects provided by political parties in QH data for the four policies

cantly higher relative coverage than BJP for the aspects [Court cases related to Aadhaar] (relative coverages : 0.11 by INC, and 0.05 by BJP), dealing with the court cases on data security and privacy issues and [Problems with Aadhaar card cancellation and enrollment centres] (relative coverages : 0.37 by INC and 0.24 by BJP), which is critical of the policy for its procedural problems. The aspects for which the currently ruling party BJP provides a significantly higher coverage than INC is [Linking of Aadhaar to various schemes] (relative coverages : 0.12 by INC and 0.23 by BJP) and [Aadhaar enrollment centres and UIDAI operations] (relative coverages : 0.19 by INC and 0.26 by BJP). The questions asked in the former aspect focus primarily on the different schemes to which Aadhaar is linked, and the problems caused to people who are unable to link Aadhaar due to different reasons. The latter discusses about the procedural issues. The aspect [Linking of Aadhaar to various schemes] does talk about the immediate issues faced by the poor after implementation of the policy. However, other aspects relevant to the poor (like [Installation of e-POS systems and Aadhaar enabled PDS] and [Opening of bank accounts for financial inclusion]) receive insignificant coverage [6.3] by both parties, when compared to the highest covered aspects, which are primarily relevant to the middle class.

4.2.1.3 GST

We find a total of 79 questions by the current ruling party BJP and 42 questions by the opposition party INC. INC provides significantly higher coverage to [Objections and confusions with GST rate slabs] (relative coverages : 0.24 by INC and 0.18 by BJP), which talks about the applicability of GST and its rates with respect to different trade sectors and [Non-conclusive discussion between centre and states regarding GST], which is a political aspect. We thus find that at the party level, INC is critical of the policy, being currently in the opposition, and one of the aspects that sees a significant difference in coverage between BJP and INC talks about the traders and companies that register for GST. Overall, we find that there is negligible coverage given to issues of the poor or the consumers by both of the parties.

4.2.1.4 Farmers protest

We find a total of 224 questions by the ruling party BJP and 166 questions by the opposition party INC. For this event, most of the aspects show nearly equal relative coverage both by BJP and INC. The only aspect where INC provides significantly more focus than BJP are [Disbursement of Loans and Subsidy by Banks for farmers] (relative coverages: 0.24 by INC and 0.15 by BJP). This aspect covers questions on problems in disbursements of loans and does not cover the structural issues related to agriculture. BJP does provide significantly higher coverage than INC to structural issues like [Farmers distress regarding minimum support price of crops]. However, the overall coverage of these aspects is negligible compared to the highest covered aspects, which do not relate to the structural issues. We thus see that despite having a much smaller number of MPs compared to BJP considering both Lok Sabha terms, INC asks a significant number of questions on each policy. INC is seen to be questioning the policies, especially regarding the procedural aspects and mechanics around their implementation. It is also evident that except for Demonetization, aspects related to the poor and the middle class, some of which address their issues in-depth, do see a coverage from the political parties. However, their overall coverage is much smaller when

compared to some of the highest covered aspects, which do not analyze the immediate problems of the poor in-depth or address them at all (for Aadhaar and Farmers Protests), or the problems of the consumers (for GST).

Chapter 5

Entity Resolution

5.1 Introduction

There could be a lot of different ways of mentioning an entity in different texts and articles. Entity Resolution is the task of disambiguating between different mentions of the same entity.

We use OpenCalais for named entity recognition (NER). It extracts entities of many different types from the articles but we only use 7 different types of entities namely Person, City, ProvinceOrState, Company, Organization, Continent. OpenCalais also provides possible resolutions from its database, they are correct mainly in the case of City, Continent and ProvinceOrState but is inaccurate for Person entities. So we have developed our own Entity Resolution algorithm for each type of entity.

Mediadb stores the data of media news articles, their metadata, entities, sentiments etc. It is a MongoDB (no-sql) database. Media data is continuously crawled from multiple news sources and stored in mediadb.

An article in the MongoDB has following fields:

- *category*: Category of the article (Regional News, National News etc.)
- *publishedTime*: Time of publishing.
- *language*: Language of the article.
- *publishedDate*: Date of publishing.
- *sourceName*: News source name.
- *country*: Country of publishing.
- *author*: Author of the article.

- *text*: Article content.
- *articleTitle*: Title of the article.
- *articleUrl*: url from which it was fetched.
- *entities*: Array of entities occurring in the article.

An entity in the MongoDB unresolved entities collection has following fields:

- *stdName*: Name of the entity
- *type*: Type of the entity (Person, Company etc.)
- *resolutions*: possible resolutions provided by OpenCalais except for Person entity.
- *aliases*: An array of all the versions and instances of the entity.
- *associatedEntities*: Co-occurring entities of any type in the article.
- *title*: Details of the Person type entity, Null for other entity types
- *articleIds*: An array of all the articles in which this entity or any of its aliases were mentioned.
- *resolved*: Flag field which is true when entity is resolved.

Graphdb stores the data of entities in the social networks. It is managed by a database management software called Neo4j. Web data from multiple sources is crawled and stored in the graphdb to build and interconnection between people, companies, locations, organizations etc.

An entity in the MediaDB unresolved entities collection has following fields:

- *name*: Name of the entity
- *uuid*: GraphDB id of the entity
- *aliases*: An array of all the versions and instances of the entity.

- *associatedEntities*: Array of interconnected entities of any type.
- *dept (optional)*: Department of IAS officers.
- *party (optional)*: Party of a politician.
- *extra (optional)*: Entity related data like title of a Person, location of Company etc.

5.2 Methodology

5.2.1 MediaDB-MediaDB Entity Resolution

The entities extracted by OpenCalais from the articles are then inserted into an unresolved entities collection. This collection will contain all the instances of an entity separately and we need to resolve all these instances into a single entity and store the entities into resolved collection which is initially empty.

For each entity in the unresolved collection, we need to find a matching entity in the resolved collection, if exists. But since there can be millions of entities in the resolved collection, comparing an unresolved entity with all the resolved entities to find the matching entity is infeasible. To achieve this, we use ElasticSearch. ElasticSearch is a highly scalable search engine which provides distributed, multitenant capable full text search engine. It provides efficient storing and querying of the data on a large scale.

We keep two copies of a resolved entity, one in the mongodb resolved collection and the second in the elasticsearch (Fig 5.1). For each unresolved entity, we query top ten matching entities by fuzzy aliases matching using ElasticSearch. From these top ten matching entities, we find the best matches. We use slightly different algorithms for finding top ten entities and best match for different types of entities. The algorithms are described below.

5.2.1.1 Finding top ten entities

To find the top 10 entities, we use match query in ElasticSearch using the attributes as described in Algorithm 2. match query uses fuzzy matching to

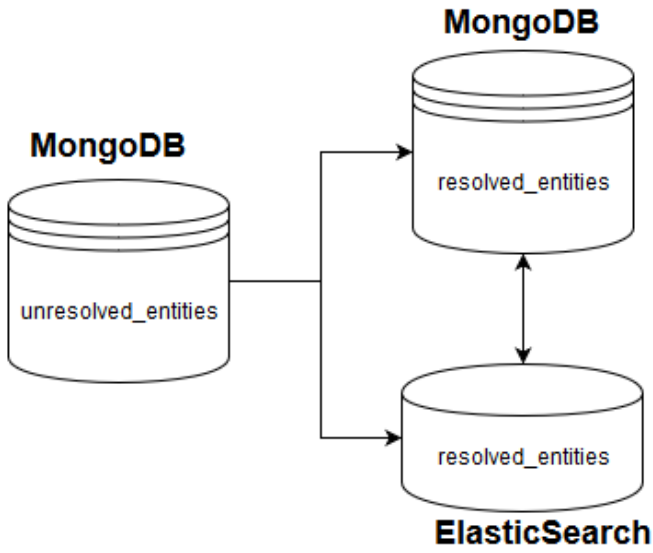


Figure 5.1: MongoDB and ElasticSearch Collections

ALGORITHM 1: Entity Resolution
INPUT: Unresolved_collection, Resolved_collection

For each entity *en* in unresolved collection
  top_ents = **FindTopEntities**$(en, Resolved\_collection)$
  $best\_match$ = **FindBestEntity** (en, top_ents)
  if $best\_match$! = NULL then
    Merge *en* and $best\_match$ and update in $Resolved\_collection$
  else
    Insert *en into* $Resolved\_collection$
  end
end

ALGORITHM 2: Finding the top ten entities (**FindTopEntities**)

Input: unresolved_entity, resolved_entities collection
Output: top_ten_entities
if *unresolved_entity.type == Person* **then**
  Filter by entities of *type = Person*
  Filter by entities that match any of the *aliases* of unresolved_entity
  Get the top ten entities that match either the *title* or *associatedEntities* (atleast one property should match) of unresolved_entity
else if *unresolved_entity.type == Company* **OR** *unresolved_entity.type == Organization* **then**
  Filter by entities of *type = Company* or *type = Organization*
  Get the top ten entities that match (any of the *aliases*(must match)) and (*resolution* (optional match))
else if *unresolved_entity.type == Country* **then**
  Filter by entities of *type = Country*
  Get the top ten entities that match (any of the *aliases*(must match)) and (*resolution* (optional match))
else if *unresolved_entity.type == Continent* **then**
  Filter by entities of *type = Continent*
  Get the top ten entities that match (any of the *aliases*(must match)) and (*resolution* (optional match))
else if *unresolved_entity.type == City* **OR** *unresolved_entity.type == ProvinceOrState* **then**
  Filter by entities of *type = City* or *type = ProvinceOrState*
  Get the top ten entities that match (any of the *aliases*(must match)) and (*resolution* (optional match))

search for the top 10 entities. It uses Levenshtein edit distance to compare the similarity between strings. For word length less than 3, it uses Levenshtein distance threshold of 0 for a valid match. For word length between 3 and 6, a distance threshold of 1 and for word length greater than 6, a distance threshold of 3. Elasticsearch uses a Boolean model for finding matching documents and also returns a similarity/relevance score calculated using a formula called the practical scoring function. This formula borrows concepts from term frequency/inverse document frequency and the vector space model but adds more-modern features like a coordination factor, field length normalization, and term or query clause boosting. Detailed explanation can be found on this link.

5.2.1.2 Finding the best match

From the top ten matching entities found by the ElasticSearch, we find the best matching entity. The results returned by the ElasticSearch query are in the order of their relevance scores. So, we start finding the best match in the same order. Whenever we find the first best match, we stop the process. If we find some best match then we merge the two entities by merging their properties/fields individually. For array properties, for e.g. aliases, associatedEntities etc., we calculate the union of both the arrays and update the property with the union in both, the resolved collection as well as the ElasticSearch collection and for stdName, we put both the names in the aliases array and store the longer name in the stdName property. If we dont find any match then we just insert this unresolved entity to the resolved collection and ElasticSearch collection. For efficiently calculating the union of two arrays, we make use of default dictionary in python. The algorithms for finding the best match are described below.

For entities of type Person, we also leverage the context information for better matching. The context information is stored in the form of associatedEntities which are the co-occurring entities in the article. But since the articles may contain a large number of co-occurring entities and some entities might not be very much relevant to the main entity, we use standard TF/IDF methods to identify the most important associated entities based on their occurrence counts in the articles. And then use this subset of associated entities in Algorithm 6.

5.2.1.3 Running on a large collection of entities

While identifying the most important associated entities based on their TF IDF values, we need to store the associated entities of all the entities in the memory. So, when we have a very large collection of entities, it might not be possible to run the algorithm directly on the complete collection as it would require large amount of memory.

So, in the case of large entities collection, we use the divide and conquer strategy for running the ER on whole collection. We divide the unresolved

ALGORITHM 3: Finding the best matching entity (**FindBestEntity**)

Input: unresolved_entity, top_ten_entities
Output: best_match
best_match = null
foreach *en* **in** top_ten_entities **do**
  if $unresolved\_entity.type == Person$ **then**
    if (**fuzzyMatchPer**(n1, n2) for every (n1, n2) **in** ($unresolved\_entity.aliases$, $en.aliases$)) **OR** (**exactMatchPer**(n1, n2) for every (n1, n2) **in** ($unresolved\_entities.aliases, en.aliases$)) **AND titleAssocMatch**(e1, e2) **then**
      best_match = en
    end
  else if $unresolved\_entity.type == Company$ **OR** $unresolved\_entity.type ==$ Organization **then**
    if $unresolved\_entity.resolution == en.resolution$ **then**
      best_match = en
    else
      if orgMatch(n1, n2) for every (n1, n2) **in** ($unresolved\_entity.aliases, en.aliases$) **then**
        best_match = en
      end
    end
  else if $unresolved\_entity.type == Country$ **OR** $unresolved\_entity.type == Continent$ **then**
    if $unresolved\_entity.resolution == en.resolution$ **then**
      best_match = en
    else
      if ($unresolved\_entity.resolution ==$ **NULL OR** $en.resolution ==$ **NULL**) **AND countryCityMatch**($unresolved\_entity.stdName, en.stdName$) **then**
        best_match = en
      end
    end
  else if $unresolved_{entity}.type == City$ **OR** $unresolved_{entity}.type == ProvinceOrState$ **then**
    if $unresolved\_entity.resolution == en.resolution$ **then**
      best_match = en
    else
      if countryCityMatch($unresolved\_entity.stdName, en.stdName$) **then**
        best_match = en
      end
    end

ALGORITHM 4: **fuzzyMatchPer**

Input: name1, name2
Output: doNamesMatch

wordList1 = list of words in name1 // "PM Narendra Modi"−> ["PM", "Narendra", "Modi"]
wordList2 = list of words in name2 // "PM Modi" −> ["PM", "Modi"]
Remove matching initials from wordList1 and wordList2 // ["Narendra", "Modi"], ["Modi"]
Remove multi − letter words (a, b) if a == b OR (doublemetaphone(a) ==
doublemetaphone(b)) OR (inital_letter_same(a, b) = true and (length(a) <
= 6 and levenshtein_dist(a, b) == 1) or (length(a)
> 6 and levenshtein_dist(a, b) == 2))
Remove an initial: I from wordList1 and multi − letter word: W from wordList2 if W
starts with I and vice − versa
if there is an unmatched element in both the lists **then**
doNamesMatch = false
else
doNamesMatch = true
end

ALGORITHM 5: **titleAssocMatch**

Input: entity1, entity2
Output: doNamesMatch

title1 = title of entity1
title2 = title of entity2
if jaro_winkler(title1, title2) < 0.88 **then**
doNamesMatch = false
else
doNamesMatch = **AssocMatch**(entity1[associatedEntities], entity2[associatedEntities])
end

ALGORITHM 6: **AssocMatch**

Input: assocEnt1, assocEnt2
Output: doNamesMatch

doNamesMatch = true
assocStr = concatenate $assocEnt1$ elements as space separated string
foreach en **in** $assocEnt2$
if fuzzywuzzy. fuzz. partial_ratio(assocStr, en[name]) < 70 **then**
doNamesMatch = false
end
end

ALGORITHM 7: **exactMatchPer**

Input: name1, name2
Output: doNamesMatch
wordList1 = list of words in name1
wordList2 = list of words in name2
if every word in wordList1 finds an exact match in wordList2 and vice − versa **then**
  doNamesMatch = true
else
  doNamesMatch = false
end

ALGORITHM 8: **orgMatch**

Input: name1, name2
Output: doNamesMatch

remove pvt|private|public|ltd|limited|inc|corp|corporation|industry|industries|enterprise **from** name1 and name2
if (name1 == name2) **OR** (name1 is an abbreviation of name2 or vice − versa) **OR** (name1 is a substring of name2 or vice − versa) **OR** jaro_winkler(name1, name2) $\geq$ 0.9 **then**
  doNamesMatch = true
else
  doNamesMatch = false
end

ALGORITHM 9: **countryCityMatch**

Input: name1, name2
Output: doNamesMatch
remove northern|southern|eastern|western|north|south|east|west **from** city name
if (name1 == name2) **OR** (name1 is a substring of name2 or vice − versa) **OR** jaro_winkler(name1, name2) $\geq$ 0.9 **then**
  doNamesMatch = true
else
  doNamesMatch = false
end

entities collection into 16 chunks. In the first level, we resolve each collection individually, to form 16 resolved collections (not resolved against each other). In the second level, we take 2 resolved chunks, merge them and then resolve the merged collection separately, to finally form 8 resolved chunks. We continue in the same manner to obtain a final resolved collection.

5.2.2 Live Entity Resolution

We also need to extract, on a live basis, the entities from the new articles which get fetched by live crawlers and run ER on the new entities. So, we have set up a cron job which gets triggered weekly and it extracts the entities from the new articles and insert them into unresolved collection and then run the ER to resolve the new entities. To identify the new articles, we use an ***extracted*** field in the article documents which is set false for the new articles. And for identifying the new entities in the unresolved collection, we use a ***resolved*** field which is set false for the new unresolved entities.

5.2.3 MediaDB-GraphDB Entity Resolution

For entity resolution between mediadb entities and graphdb entities, we first create the unresolved collection from graphdb, which is then resolved against the resolved mediadb entities collection. We extract the entities data from Neo4j and insert into unresolved collection of MongoDB by changing the labels of GraphDB entities fields with their corresponding MediaDB fields as described in 5.1.

There is only a slight change in the algorithm of GraphDB to MediaDB resolution compared to MediaDB to MediaDB resolution. In Algorithm 6, we change the threshold to 65 instead of 70 because the associated entities of media entities are mostly different from that in graphdb entities.

Graph DB	Media DB
Politician	
Name, Aliases, Party	StdName, Aliases, Associated Entities
Bureaucrats	
Name, Aliases, Position, Department	StdName, Aliases, Title, Associated Entities
Businessperson	
Name, Aliases, Company	StdName, Aliases, Associated Entities
Company	
Name, Ownership Information	StdName, Associated Entities
Ministries	
Name, Aliases, Location	StdName, Aliases, Associated Entities
City	
Name, Aliases, State	StdName, Aliases, Associated Entities

Table 5.1: The properties of Graph DB and media entities used for ER between graph and media data

5.3 Evaluation

To evaluate the entity resolution algorithm, we took a random sample of 100 entities into unresolved_collection and created a gold standard of 100 randomly selected entities. After resolution, we take each pair of entities and classify them into following four classes:

- True Positive (TP) - If the pair exists in both gold standard and resolved collection.
- False Positive (FP) - If the pair does not exist in gold standard but exists in resolved collection.
- False Negative (FN) - If the pair does exists in gold standard but not in resolved collection.

Precision	89%
Recall	81%
F1 Score	85%

Table 5.2: MediaDB-MediaDB Resolution

Precision	97%
Recall	89%
F1 Score	93%

Table 5.3: GraphDB-MediaDB Resolution

- True Negative (TN) - If the pair does not exists in both gold standard and resolved collection.

Now, we use the following formulas for evaluation:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Table 5.2 shows the ER accuracy for MediaDB to MediaDB entities resolution, and Table 5.3 shows ER accuracy for GraphDB to MediaDB entities resolution.

Chapter 6

Deep Learning Classifiers

6.1 Introduction

Researchers have leveraged various sentiment Analysis tools like *Sentistrength, Vader, Alchemy*, etc. for analyzing the sentiment exhibited in text pieces. However, these tools are simply based on the presence or absence of certain words which are clearly indicative of certain sentiment, for example, simple speaking, 'good' counts for positive sentiment and 'bad' counts for negative sentiment. Many times, they ignore the sentence structures, negation and contextual information which also play a key role in determining the overall sentiment of the statement. Most of the conventional techniques essentially focus on the bag of words models which ignore syntax and also cause neutralization across sentences.

Deep Learning models for Natural Language Processing have proved to be effective in incorporating the complicated nuances of language and therefore predict the correct sentiment. Iyyer et al. (2014) have used recursive neural networks (RNNs) for the sentential level political ideology detection. Inspired from their work, we build Recursive Neural Network models to develop two classifiers, **Technology Determinism vs Skepticism** and **Pro vs Anti Policy**.

Recursive Neural Networks are a type of hierarchical neural network which take into account both the syntactic and semantic features of the sentence. The assumption of the algorithm is that the meaning of each phrase should be a combination of the meaning of the words in it applied recursively based on the syntax. The relation is given in terms of the equation below

$$x_p = f(W_L.x_a + W_R.x_b + b_1)$$

where x_a, x_b at the ground level are the word embeddings of the words derived from a embedding matrix W_e of dimension $d \times V$(V is the size of the vocabulary) and f is non-linear function and W_L, W_R, b_1 are all parameters of dimensions $d \times d, d \times d, d \times 1$ and x_p is the vector representation of the phrase. The Ideology of each phrase is then calculated using the softmax function as

$$\hat{y_p} = softmax(W_{cat}.x_p + b_2)$$

Again W_{cat}, b_2 are parameters of dimension $2 \times d, 2 \times 1$.

6.2 Technology Determinism vs Skepticism

We develop a deep learning classifier that would identify a given statement as pro-technology (Technology Determinism) or anti-technology (Technology Skepticism). Pro Technology statements generally would show faith and support for technology in solving many problems while the Anti statements would show doubt and skepticism about using technology. Some examples of Pro and Anti-technology statements have been provided in Appendix B.1

6.2.1 Dataset Development

- We have developed a **dataset of 850 statements** relevant to the discussion about the role of technology in various affairs. The By statements have been extracted from the crawled news articles using certain keywords that are common in Technology-Related discussions.
- **Manual Annotation -** With the help of all people in the research group, this has been manually labeled as Pro/Anti. The ambiguous cases have been resolved using the context information.
- **Context Information -** We preserve the context information while extracting statements. So, we store the preceding and succeeding statements for each by statement in the format ***preceding statement;; by***

statement;; succeeding statement. We are using this context information to resolve ambiguous cases in the manual annotation.

For example, for the sentence, The technology has undergone a drastic transformation in the last 20 years, Modi said adding the aspirations of the youths have to be kept in mind in this era, the following is the context information being stored using the script. The PM said India's economy is being transformed and the manufacturing sector is getting a boost.;; The technology has undergone a drastic transformation in the last 20 years, Modi said adding the aspirations of the youths have to be kept in mind in this era.;; On merits of democracy, the Prime Minister said, Bigger than the strength of the government is the people's power.

So, if we consider the words, boost in the preceding statement and the effect of technology on the economy and manufacturing sector, the statement appears to be Pro Technology.

- **Pure Statements -** The raw statements generally have phrases like "Mr. Modi said,", "Chief Minister Manohar Parrikar on Thursday said that", "Parrikar, before leaving the meeting venue at Dona Paula, said,". For our classifier to work effectively, we need to drop these kinds of phrases and preserve only the statement said by a particular entity. So, along with the annotation task, we have also rewritten the pure statement after dropping above kind of phrases from the raw statements.

6.2.2 Parsing

We are supposed to convert an input sentence into the **parse tree** to feed it as input into the **Recursive Neural Network model**. Iyyer et al. (2014) depict using the following figure [6.1] how a sentence is decomposed into its parse tree and the model then works bottom-up assigning each phrase a label and incorporating the phrase labels with learned weights to predict the label of the original sentence.

So, we have implemented the code for converting a given sentence into the **parse tree representation**. We are leveraging the **Stanford Parser** and

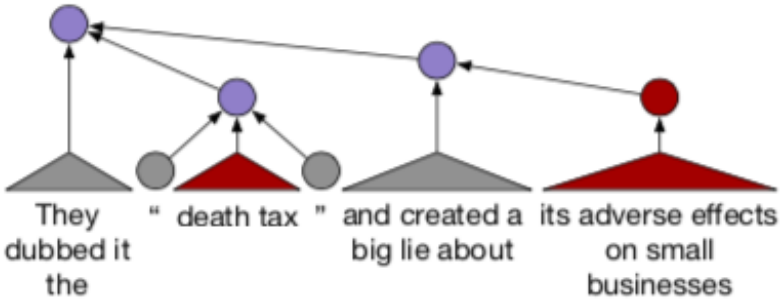


Figure 6.1: An example of compositionality in ideological bias detection (red → conservative, blue → liberal, grey → neutral) in which modifier phrases and punctuation cause polarity switches at higher levels of the parse tree. (Figure from Iyyer et al. (2014)[7])

NLTK library for this purpose. First, we convert the sentence into **Part-of-speech tagged text** which is then converted into the **binary parse tree**. For example, for the sentence, We, as a government, are taking help of every technology to take benefits of treatment to poor and in rural areas, even using cloud computing, the PoS text and the parse tree can be found in the appendix B.3.

6.2.3 Training & Implementation

The in-house dataset that was developed and annotated has a total of 853 statements, out of which 769 are pro and 84 are anti. This corpus has a total of 21262 words with a vocabulary size of 3927. We trained the dataset for **1000 epochs** on Microsoft Azure VM. **Tanh** has been used as the non-linear activation function. The root nodes in the parse tree have been assigned the label of the sentence (pro: 0, anti: 1). For the new words that have not been seen by the model, UNK (unknown) token is assigned to them. We use the **cross-entropy loss function** for the backpropagation algorithm. We make a **9:1 train vs validation split** after randomly shuffling the entire dataset. The parse trees are generated using the Stanford Parser &

NLTK library after which they have been stored as pickle objects to speed up the successive implementations. We have also used **L2 regularization** to prevent overfitting. We are using the **Stochastic Gradient Descent Optimizer** with a learning rate of 0.01. Also, we are clipping the gradients to prevent them from exploding. We have built the deep learning classifier leveraging the Pytorch framework. The architecture of the model consists of 4 layers, 1st embedding layer followed by a linear layer which is succeeded by a non-linear tanh layer and finally the linear projection layer. The embedding dimension used is 300.

6.2.4 Results

After training the model, we have evaluated it on both the training set and validation set. The results are as following

- **Training Accuracy: 99%**
- **Validation Accuracy: 95%**

Although the 95% validation accuracy seems impressive but actually due to the heavy class imbalance the majority class prediction baseline is itself at 90%.

6.2.4.1 Class Imbalance & Undersampling

Because of the highly skewed dataset, we also took a smaller subset of the dataset with an equal number of pro and anti statements (84 each). We trained the Recursive neural network model on this dataset in a similar way as on the original dataset for 1000 epochs. The results are as following

- **Training Accuracy: 100%**
- **Validation Accuracy: 81%**
- **Validation (All dataset) Accuracy: 91%**

- **Pro Class: Precision 0.86 Recall 0.75 F1 score 0.8**
- **Anti Class: Precision 0.77 Recall 0.88 F1 score 0.82**

Unlike the complete dataset, here we see a big jump to 81% from the random/majority class prediction baseline which is at 50% in this case. The model has a quite good performance provided the limitation of the dataset size. We have also validated on the entire ground truth dataset. Here, we got an indeed better validation accuracy of 91%. This is because the model is more inclined towards labeling statements as Pro than Anti in confusing situations and in the case of the entire ground truth dataset as validation set because of the sheer abundance of the pro statements, the accuracy is bound to increase.

6.2.5 Misclassifications

We also analyzed the statements which have been wrongly classified. Some of them are as following :

6.2.5.1 Actually Anti but classified as Pro

1. Given the history of abuse by governments, it is right to ask questions about surveillance, particularly as technology is reshaping every aspect of our lives.
2. Though the country is spending huge amounts to import modern weapons from other countries, what we get most of the times are outdated technology systems.
3. Describing the Islamic State as one of the best users of Internet technology", has asked the armed forces to be prepared for future cyberwars while equipping soldiers for the physical battlefield.

6.2.5.2 Actually Pro but classified as Anti

1. Science is universal, but technology has to be local.

2. The commissioning of the facility also symbolizes the country's capability in establishing such world-class facilities wherein technology from outside is restricted or not available.

So, we observe that most of the misclassified statements are either themselves ambiguous (i.e. its a bit confusing to know their gold label otherwise as well), or they have complicated sentence structures where one phrase is anti while the other is pro or they have some words which are generally representative of some sentiment e.g. restricted is generally a negative word and therefore the last sentence is being classified as Anti.

6.2.6 Word2Vec Embeddings

Google provides learned word2vec representations for a large vocabulary of words. These are basically the vector representations corresponding to each word. Words having similar meanings are expected to have close word vectors. Google trained the vectors on a part of Google News dataset (about 100 billion words). The model contains 300-dimensional vectors for 3 million words and phrases.

We have tried to initialize the embedding layer of the Recursive neural network with these pre-trained word2vec embeddings. By default, all the word embeddings are randomly initialized and the model is allowed to learn them as part of the training process. However, initializing with word2vec embeddings and then learning on top of them provides us a 1-2% increase in accuracy at times. Word2Vec initialization helps as it is a good starting point for the embedding layer compared to the arbitrarily random initializations.

6.3 Pro vs Anti Policy

We have built a classifier for identifying the stance of statements by eminent people on various important policies. Pro statements are the ones where the speaker is in support of the policy or appreciates the policy. In Anti statements, the speaker is found to criticize the policy or the speaker talks

Policy	Number of Statements
Aadhar	659
Demonetisation	410
GST	544
Farmers Protests	791
Total	2404

Table 6.1: Number of by statements for each policy

Stance	Percentage of Statements
Pro	44%
Anti	18%
Neutral	38%

Table 6.2: Percentage of statements classified among each of the three classes after the annotation

of its drawbacks whereas Neutral Statements are the ones where there is no certain stance, this might happen when the statement is factual or otherwise. Some examples of Pro and Anti-Policy statements have been provided in Appendix B.2.

6.3.1 Dataset Development

The dataset for the classifier includes By statements from 4 important national policies namely, **Aadhar, Demonetisation, GST and Farmer Protests.** (Table 6.1)

To train the classifier, statements have been annotated as either Pro, Anti or Neutral. The Raw statements have been rewritten as pure statements after dropping the phrases like "Mr. Modi said,", "Chief Minister Manohar Parrikar on Thursday said that" etc. Also, the preceding and succeeding statements were provided for the annotation task as they might be helpful to get a sense of the context. Duplicate and irrelevant statements were deleted while cleaning the dataset.

Table 6.2 depicts the percentage of statements classified among each of the three classes after the annotation.

Training Accuracy	99%
Validation Accuracy	83%
Precision	85%
Recall	61%

Table 6.3: Neutral vs Position Classifier

Training Accuracy	99%
Validation Accuracy	80%
Precision	78%
Recall	98%

Table 6.4: Pro vs Anti Classifier

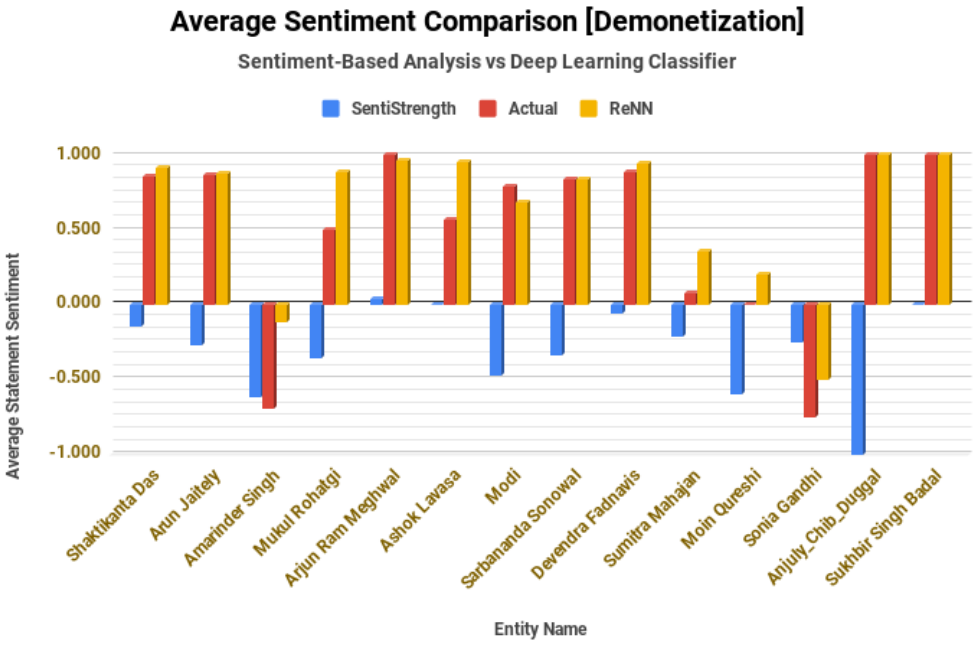
The entire corpus has 46634 words found with a vocabulary size of 6281. The Recursive Neural Network Model has been built analogously as to the model built for the Technology Determinism vs Skepticism Classifier.

6.3.2 Results

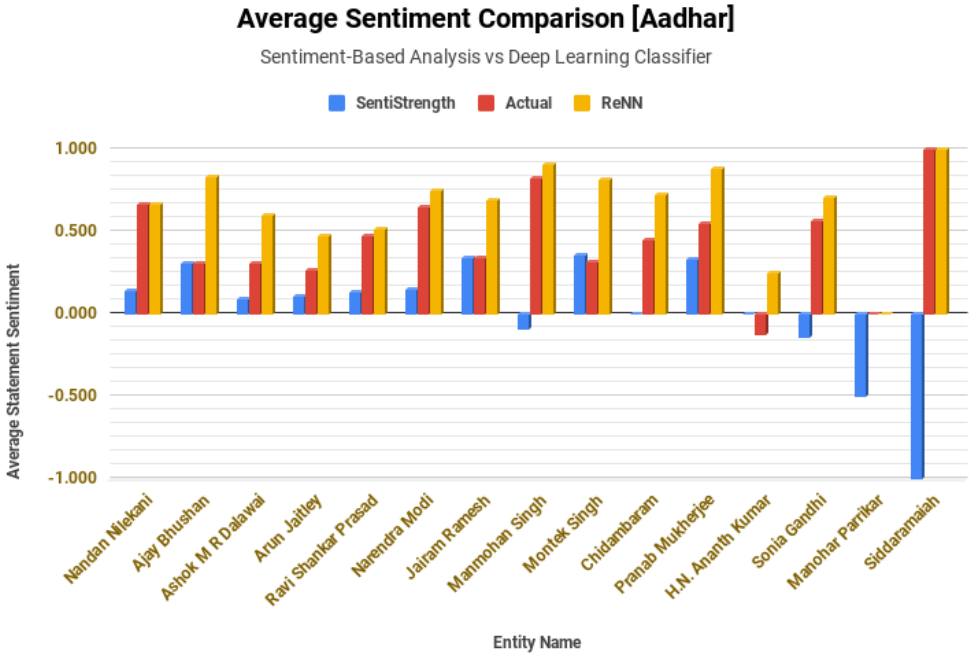
However, here we build two classifiers, first Neutral vs Position classifier (Table 6.3) and second Pro vs Anti classifier (Table 6.4). The second classifier is used in case the first classifier identifies a statement as having a position instead of being neutral.

6.4 Sentiment Analysis Vs Deep Learning Classifiers

As we know, sentiment analysis tools like Sentistrength work essentially like a bag of words model where they try to basically count on the presence and absence of certain sentiment-bearing words. We show in Table 6.5 that our deep learning classifier is able to outperform Sentistrength on several occasions.

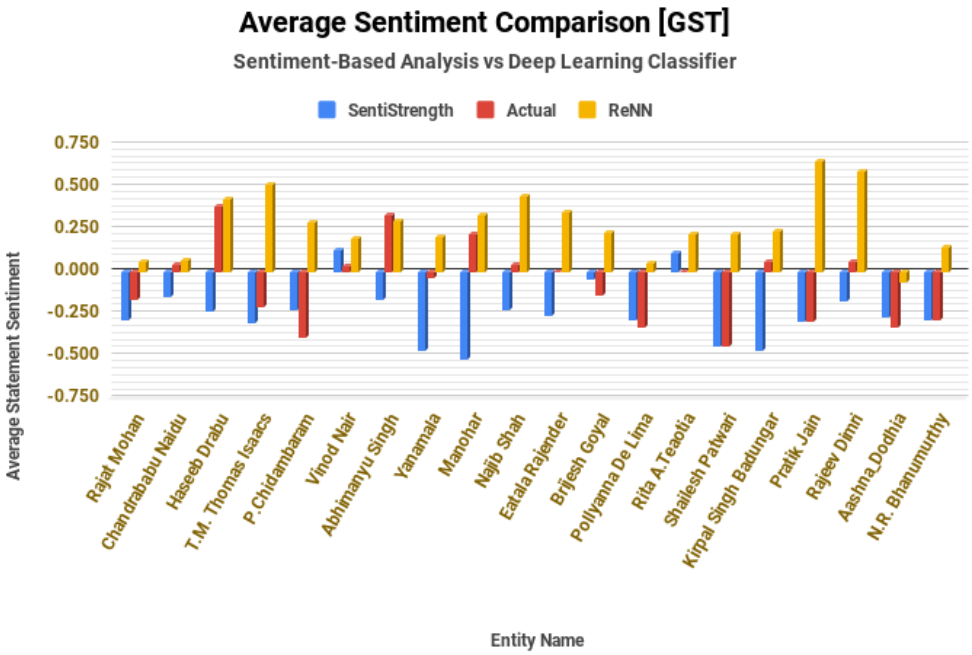


(a) Demonetization

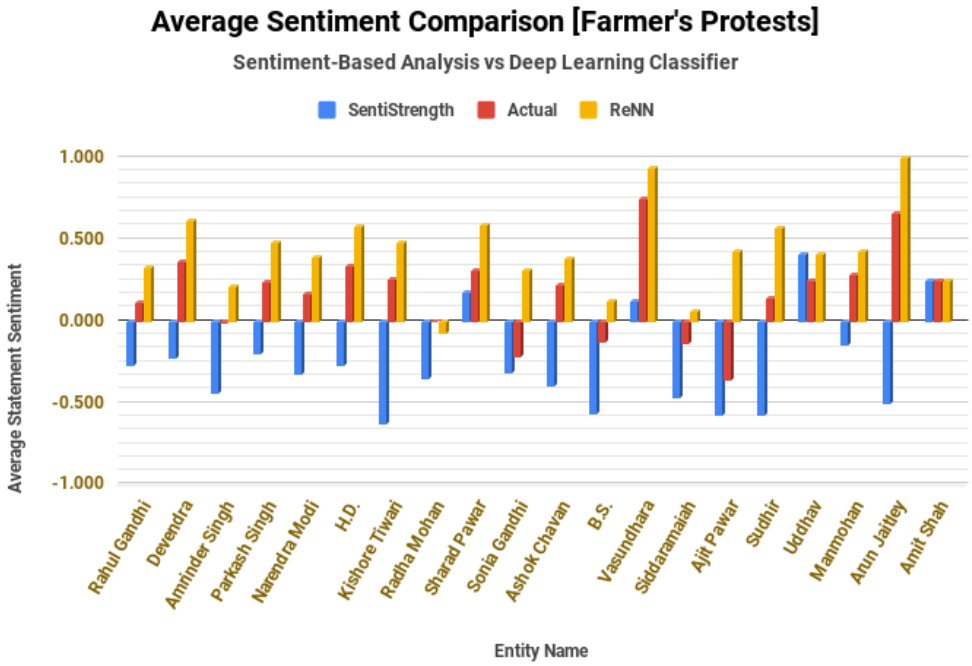


(b) Aadhar

Figure 6.2: Average Sentiment comparison between top entities for the four policies

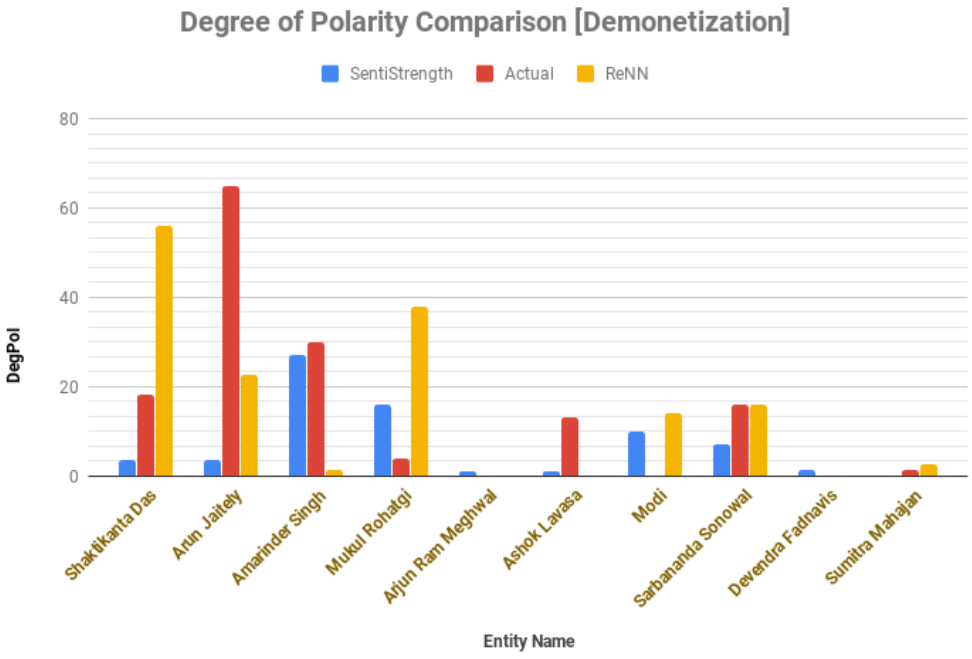


(c) GST

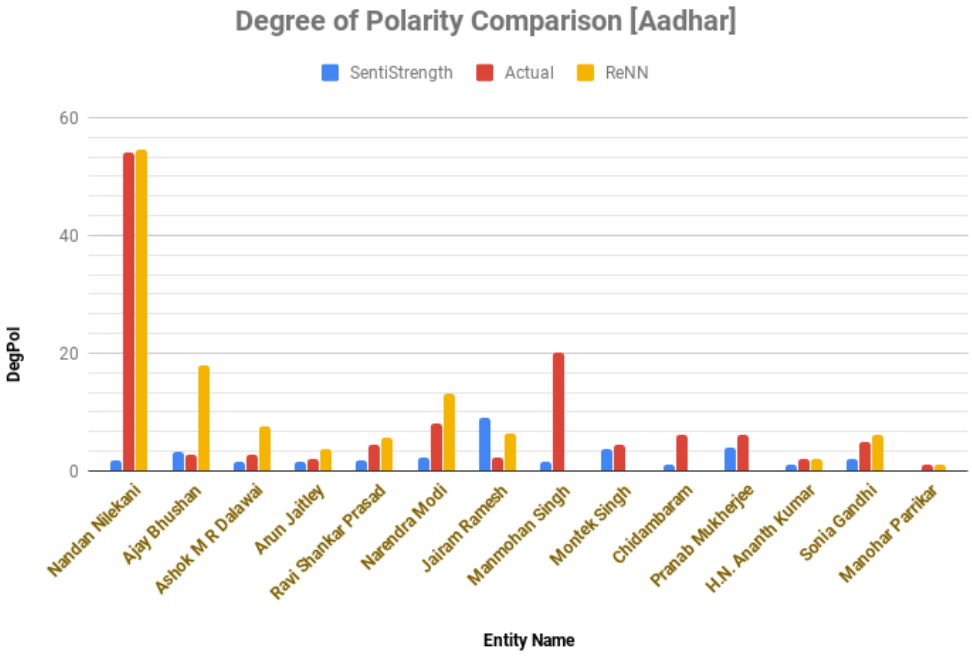


(d) Farmers' Protest

Figure 6.2: Average Sentiment comparison between top entities for the four policies

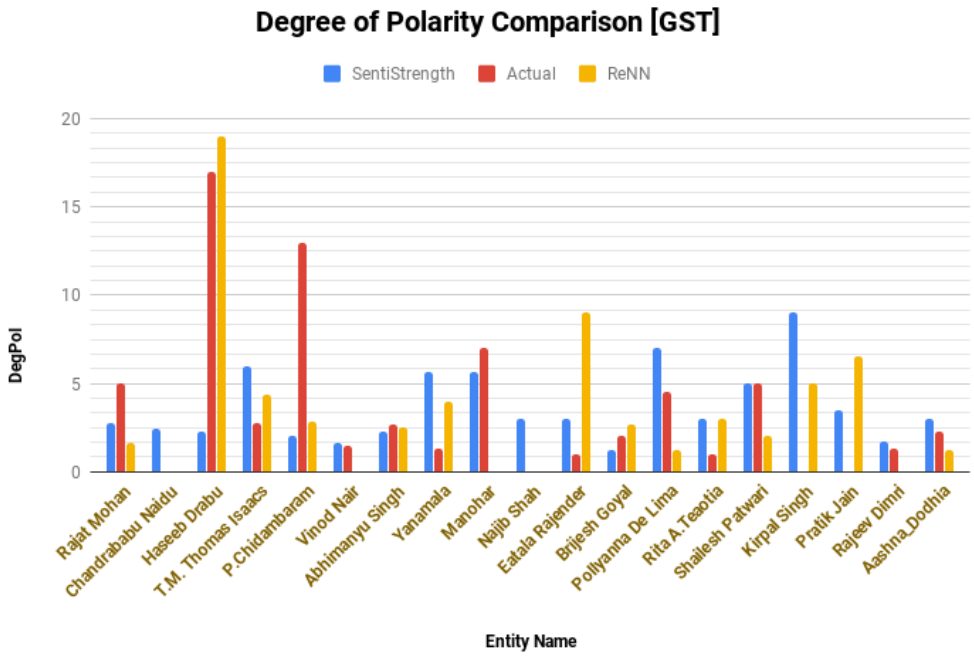


(a) Demonetization

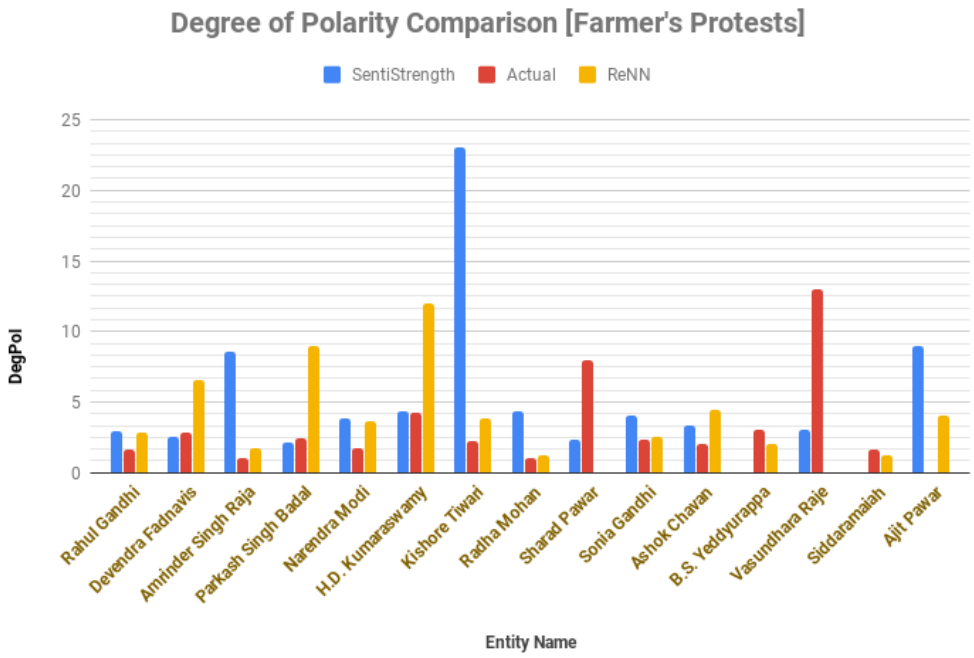


(b) Aadhar

Figure 6.3: Degree of Polarization comparison between top entities for the four policies



(c) GST



(d) Farmers' Protest

Figure 6.3: Degree of Polarization comparison between top entities for the four policies

Entity Statement	Classifier Stance	Senti Strength Score	Explanation for SentiStrength Failure
In order to help children suffering from an e-learning tool has also been developed by the National Institute of Mentally Handicapped and all the scholarships will be under one	Pro	-3	Mentions word Autism
Bangalore has enormous energy and ideas to solve problems.	Pro	-1	Mentions word problems
At UIDAI, we are very strict on privacy issues.	Pro	-2	Mentions word strict
Earlier farmers used to get insurance of Rs 50,000 on death or permanent disability under Raj Sahakar Personal Insurance Scheme, which now has been increased to Rs 10 lakh.	Pro	-2	Mentions words, death and disability
Prime Minister Narendra Modi 's grand MSP increase for farmers is like applying a band-aid to a massive hemorrhage.	Anti	1	Sarcasm
Just because it is possible to hack a network did not mean that technology must not be deployed.	Pro	-1	Negation
Aadhaar would turn into a boon for marginalized or vulnerable groups to get access to many services with the help of single identity number opined.	Pro	-1	Mentions words like weak & marginalised sections
In the long term, this landmark step will increase the size of the official economy and reduce the shadow economy	Pro	0	Has both the words increase and reduce
The Pradhan Mantri Jan-Dhan Yojana provides a platform for changing the economic condition of our people.	Pro	0	Mentions 'change' rather than direct positive words like 'better'

Table 6.5: Deep learning classifier vs SentiStrength

Chapter 7

Conclusion and Future Work

We have tried to perform a comprehensive analysis of various important national policies and events. In the process, we have developed a framework of certain key steps like Entity Resolution, Topic Modeling, Sentiment Analysis and Stance Detection. For entity resolution, we have highly optimized the existing functioning and also set-up a live entity resolution mechanism. We have leveraged state-of-the-art natural language processing and machine learning techniques. And for the stance detection, we have also used deep learning to build neural network classifiers.

There are definitely some interesting directions to be pursued following the results we have achieved.

7.1 Tree LSTMs

Recursive Neural Networks sometimes fail to detect the change in stance in the tree structure. To facilitate this, TreeLSTMs [5] should be tried to better learn when to ignore the information of the sub-tree. TreeLSTMs are another kind of deep neural network that incorporates the hierarchical nature of sentences. TreeLSTM is a variant of ReNN where each node unit is an LSTM cell. There is a forget gate for each child of a node. Some papers claim to achieve better results using TreeLSTMs than those found by using recursive neural networks. Applying TreeLSTMs to our domain is definitely one interesting idea to work upon.

7.2 Balanced Binary Trees

Also, it would be worth trying changes in the structure of the trees while incorporating Balanced Binary Trees [6] instead of the syntax trees. Balanced

Binary Trees are constructed by building a balanced binary tree while using the words as leaves. They can make the trees more shallow so that the model can learn quickly and better. As mentioned in (Shi et al., 2018)[6] these trees give almost the same and even slightly better than syntax trees on some benchmarks.

7.3 Phrase-Level Labelings

The Recursive neural network model can be trained while incorporating the phrase level annotations. Although developing such a dataset is a challenge but (Iyyer et al., 2014)[7] show that phrase-level annotations improve the performance of the model. This is expected because in this case, the model is able to learn bottom-up from the phrase-level stance labels.

7.4 Larger Dataset

The deep learning model performs quite well despite the limitation of the size of the dataset. However, it is certainly a promising idea to train the models over a much larger dataset as that would be able to truly unleash the powers of deep neural networks. It might require crowdsourcing resources for being able to develop the ground truth for such a large dataset.

7.5 Unstructured Datasets

It sounds very interesting to extend the idea to **unstructured datasets**. These days, social media has become a prominent platform for all kinds of discussions including the political & economic discussions. However, there are differences in the type of statements found in mass media to the ones found on Twitter. These differences have to be adequately addressed in the approach and consequently be learned by the model.

7.6 Other Tweaks

Along with the above one could also explore some little tweaks in Iyyer et al. (2014)[7] such as using tree representation which is the representation of root concatenated with the average of the rest of the nodes in the tree instead of just root representation. Also, Iyyer et al. (2014)[7] report better accuracy by initialization of weight matrix to I/2, i.e, giving equal weight to each child of nodes initially. Moreover, it makes sense to try feature engineering. Some features like Part of speech tags are worth including in the list of experiments to be pursued.

Appendix A

LDA Analysis on Aadhar Media Corpus

LDA Topic Number	Aspect
Topic#0	PDS & Aadhar Linking
Topic#3	Elections, Voter Identities & Aadhar
Topic#4	Digitization, Cashless Payments
Topic#7	Official Documents like Passport, Birth Certificates etc.
Topic#9	Aadhar Act, Indian Economy & Policy
Topic#10 & Topic#12	Aadhar Enrollment
Topic#13	Measures by UIDAI
Topic#14	Crimes & Aadhar Card
Topic#15	Pensions
Topic#17	Governance, Policy & Legislation
Topic#20	Details on Aadhar Cards & Distribution of Aadhar Cards
Topic#21	Women, Minority Groups & Aadhar Registrations
Topic#22	Farmers
Topic#23	Opposition & Politics over Aadhar
Topic#24	Linking of Aadhar with other schemes
Topic#25	Court & Controversy Over Aadhar Card
Topic#26	Banks & Aadhar Cards
Topic#27	Misc.
Topic#29	LPG Subsidy & DBT (Direct Benefit Transfer)
1,2,5,6,8,11,16,18,19,28	Zero Articles or Irrelevant Topics

Figure A.1: LDA Analysis over Aadhar Media Corpus

Appendix B

Deep Learning Classifiers

B.1 Pro & Anti Technology Statements Examples

B.1.1 Technology Determinism (PRO)

1. Artificial Intelligence (AI), the theory of enabling computer systems to perform tasks and make decisions normally done by human beings, will help in eradicating poverty and disease, Prime Minister Narendra Modi said on Sunday.

2. Mr. Modi said he had placed science and technology at the forefront of the countrys diplomatic engagement.

3. Mr. Modi told media leaders that digital technology can help in innovation, empowerment, and democratization.

4. Prime Minister Narendra Modi on Friday asked BJP MPs to do their utmost to use mobile technology to get in touch with the youth.

5. Modi also said that in India, the government is using technology for accountability and transparency.

6. People want to join BJP and there should be no single BJP primary member whose info is not available online," Modi said, adding that BJP should even use technology for internal communication".

7. Noting that India had carried out nuclear tests on Buddha Purnima on May 11 in 1998, when BJP stalwart Atal Bihari Vajpayee was prime minister, Modi said youths need to imbibe Vajpayee's mantra of 'Jai Vigyan' (hail science) to make India modern and strong.

8. "The Centre will fully stand by the Andhra government in this hour of difficulty," Modi said, praising the IMD for correctly forecasting on October 6 the velocity, direction and time of the cyclone, leveraging the power of technology.

9. Information and communication technology (ICT) enabled courts will ultimately help in moving to the next level of full-fledged e-courts," Mukherjee said at the concluding ceremony of the 150th anniversary of Calcutta high court at Netaji Indoor Stadium on Sunday.

B.1.2 Technology Skepticism (ANTI)

1. Today, we misuse technology and kill girls in the womb of the mother, Modi said, adding that the damage being done through generations would take another two to three generations to be rectified.

2. Meanwhile, A Sai Manohar, senior superintendent of police, Indore, says, "Criminals adopt technology faster than law enforcement agencies.

3. Lucknow: In a veiled attack on those demanding video proof of the surgical operation across the LoC, minister Manohar Parrikar on Thursday said that technology is so advanced that video footage can be edited to suit one's convenience.

B.2 Pro & Anti Policy Statements Examples

B.2.1 Pro Policy Statements

- For the marginalized , illiterate and vulnerable groups , Aadhaar will come as a boon to would turn into boon for marginalized or vulnerable groups to get access many services with a help of single identity number opined.

- Aadhaar project is an example of using modern technology to leapfrog for future development and transformation of a country
- The government saved about Rs 50,000 crore LPG subsidy due to the linking of Aadhaar card with Jhan Dhan accounts.

B.2.2 Anti Policy Statements

- The Central government is eating up all resources and the states have been reduced to the level of dignified municipalities.
- The situation was turning from bad to worse with people suffering for non-availability of cash.
- This is because despite waiver , banks have still not started disbursing fresh credit to the farmers leaving them fund starved.

B.3 PoS Text and Parse Tree

Original Sentence:

We, as a government, are taking help of every technology to take benefits of treatment to poor and in rural areas, even using cloud computing.

Part-of-Speech Tagged Text:

```
(ROOT
 (S
  (NP
   (NP (PRP We))
   (, ,)
   (PP (IN as)
    (NP (DT a) (NN government)))
   (, ,))
```

```
(VP (VBP are)
  (VP (VBG taking)
    (NP
      (NP (NN help))
      (PP (IN of)
        (NP (DT every) (NN technology)
          (S
            (VP (TO to)
              (VP (VB take)
                (NP
                  (NP (NNS benefits))
                  (PP (IN of)
                    (NP (NN treatment))))
                (PP
                  (PP (TO to)
                    (NP (JJ poor)))
                  (CC and)
                  (PP (IN in)
                    (NP (JJ rural) (NNS areas))))
                (, ,)
                (S
                  (ADVP (RB even))
                  (VP (VBG using)
                    (NP (NN cloud) (NN computing)))))))))))))
```

Appendix C

Coreference Resolution Results

S.No	Article	Coreference Results
1	GANDHINAGAR: State-level workshop to understand various types of digital payment system was organized at Gandhinagar in the presence of Gujarat chief minister Vijay Rupani . In this context, Rupani expressed his commitment that Gujarat will be leading in cashless economy and digital banking transactions."With a commitment to begin an era of honesty in the nation, our Prime Minister Narendra Modi has taken a brave step in demonetization," the CM said."Gujarat will adopt the cashless economy path through various mediums such as SMS banking, e-wallet, digital payment, Aadhar seeding, etc.," Rupani said.Rupani appealed to government employees and various departments to adopt digital transaction practices and encourage people of Gujarat to do the same. Officials of nationalized banks, including State Bank of India , and officers and karmyogis of Sachivalya were also asked to attend the workshop.	CHAIN1-["GANDHINAGAR" in sentence 1, "Gandhinagar" in sentence 1] CHAIN12-["Gujarat chief minister Vijay Rupani" in sentence 1, "Rupani" in sentence 2, "his" in sentence 2] CHAIN28-["Gujarat" in sentence 1, "Gujarat" in sentence 2, "Gujarat" in sentence 4, "Gujarat" in sentence 4]

2	AHMEDABAD: Hailing Prime Minister Narendra Modi 's decision to demonetize Rs500 and Rs 1,000 notes, Union minister of textiles Smriti Irani said that the move is part of the fight against black money. "People queuing up outside banks are part of the fight against black money and I salute their spirit," said the Rajya Sabha MP from Gujarat.Irani inaugurated a thematic exhibition of handicrafts in Ahmedabad on Saturday and also distributed 'Pehchan' identity cards to artisans from five clusters in Gujarat, namely Jamnagar, Naroda, Surendranagar, Amreli and Kalol. Irani had launched the 'Pehchan' initiative last week at Sant Kabir Nagar in UP, which is a move to register and provide ID cards to handicraft artisans and link them to a national database. "The upgraded ID cards for artisans will be linked to their Aadhar card numbers and bank accounts so they can receive all cash transfer benefits from various government schemes directly," said Irani.In a bid to encourage women and BPL artisans in metal craft, the Union minister for textile also announced that tool kits and safety equipment would be given to them for free for their respective crafts."In February, the ministry will also organize insurance camps where artisans live and provide them a credit guarantee scheme and insurance at their doorstep," said the minister.	CHAIN16-["AHMEDABAD" in sentence 1a, "Ahmedabad" in sentence 2] CHAIN80-["the Union minister for textile" in sentence 4, "the minister" in sentence 5] CHAIN49-["ID cards" in sentence 3, "them" in sentence 3] CHAIN40-["Smriti Irani" in sentence 1, "Irani" in sentence 3] CHAIN28-["the fight against black money" in sentence 1, "the fight against black money" in sentence 2] CHAIN78-["artisans" in sentence 4, "their" in sentence 4, "they" in sentence 4, "them" in sentence 4, "their" in sentence 4, "them" in sentence 5, "their" in sentence 5]

3	NEW DELHI: Former UIDAI chief Nandan Nilekani has been roped in by the BJP government as a special invitee in the 13-member committee of chief ministers, representing different political outfits, set up to promote digital payment systems, mainly in rural areas after the Centre scrapped high-value currency notes.The appointment of Infosys co-founder and face of UPA government's flagship 'Aadhaar' programme into a government panel came as a surprise as Nilekani resigned as UIDAI chief to contest 2014 Lok Sabha polls on a Congress ticket. Nilekani contested from Bangalore South and lost to BJP's Ananth Kumar who is parliamentary affairs minister in Modi government.The panel, headed by TDP boss and Andhra Pradesh CM Chandrababu Naidu , also has BJD chief and Odisha CM Naveen Patnaik, chief minister of MP Shivraj Singh Chouhan, Sikkim CM Pawan Kumar Chamling and V Narayanasamy, who is heading Congress government in Puducherry as members. Maharashtra CM Devendra Fadnavis is also a member of the committee along with NITI Aayog vice-chairman Arvind Panagariya. However, Bihar chief minister Nitish Kumar, who has backed Prime Minister Narendra Modi's demonetisation move and irked opposition parties, is not on the panel.Other special invitees in the committee are Janmejaya Sinha, chairman, Boston Consulting Group, Rajesh Jain,	CHAIN32-["Former UIDAI chief Nandan Nilekani" in sentence 1, "Nilekani" in sentence 1, "Nilekani" in sentence 2] CHAIN34-["BJP" in sentence 1, "BJP 's" in sentence 2] CHAIN83-["Sharad Sharma" in sentence 4, "UPI" in sentence 5] CHAIN6-["UIDAI" in sentence 1, "UIDAI" in sentence 1] CHAIN39-["Congress" in sentence 1, "Congress" in sentence 2] CHAIN55-["the 13-member committee of chief ministers" in sentence 1, "the committee along with NITI Aayog vice-chairman Arvind Panagariya" in sentence 3, "the committee" in sentence 4]

	managing director, net-CORE, Sharad Sharma, cofounder, iSPIRIT and Jayant Varma, professor (finance), IIM (Ahmedabad). The high-powered panel is mandated to identify global best practices for implementing an economy primarily based on digital payment and examine the possibility of adoption of these global standards in the Indian context.The panel will also outline measures for rapid expansion and adoption of the system of digital payments like cards (debit, credit and prepaid), digital-wallets/ e-wallets, internet banking, unified payments interface (UPI), banking apps, etc and shall broadly indicate the roadmap to be implemented in one year.The Naidu-led panel will also prepare an action plan to reach out to the public at large with the objective to create awareness and help them understand the benefits of such a switch-over to digital economy and come out with a roadmap for the administrative machineries in the states to facilitate adoption of digital modes of financial transactions.	CHAIN90-["a government panel" in sentence 1, "The high-powered panel" in sentence 5, "the Indian context.The panel" in sentence 5]$$$

4	MUMBAI: Sep 20 , 2015, DHNS: 1:48 ISTMumbai-based activist obtains information from UIDAIContracts for Aadhaar card project undertaken by the Unique Identification Authority of India (UIDAI) were given without proper tendering process by the then Congress-led UPA government at the Centre, a reply obtained under the Right to Information Act (RTI) has revealed.Mumbai-based RTI activist Anil Galgali has obtained the information from the UIDAI that was once headed by Infosys co-founder Nandan Nilekani.UIDAI Public Information Officer S S Bisht informed Galgali that for Aadhaar card, work has not been issued any kind of tender.A total of 25 companies have been awarded different responsibilities. The empanelment of agencies is being done under empanelment process guidelines contained in RFE (Request for Empanelment) dated 19th May 2014, he said.Another UIDAI Public Information Officer and Deputy Director R Harish informed Galgali that Rs 13,663.22 crore was the approved outlay for UIDAI project and Rs 6,562.88 crore amount incurred up to date May 31, 2015, since inception.PM requestedGalgali has requested Prime Minister Narendra Modi to personally look into the issue to bring in more transparency and get the previous contracts reviewed and investigated so that all questions are settled.This is very unfair where the project relates with the sensitive	CHAIN66-["Galgali" in sentence 1, "Galgali" in sentence 2, "Galgali" in sentence 2, "RTI" in sentence 2, "HCL" in sentence 2, "Tata Consultancy" in sentence 2, "SQTC" in sentence 2, "BSNL" in sentence 2] CHAIN92-["Aadhaar card project undertaken by the Unique Identification Authority of India -LRB- UIDAI -RRB- were given without proper tendering process by the then Congress-led UPA government at the Centre" in sentence 1, "the project" in sentence 2] CHAIN63-["Linkwel Telesystem" in sentence 2, "NISG" in sentence 2]

	data of 125 crore Indians of the country and no tendering process was done, said Galgali in a press statement on Saturday.8 contracts awardedThe RTI query has revealed that a total of eight contracts was awarded to Wipro and HCL, two contracts to Tata Consultancy, 14 contracts to Mac Associates, HP (India) Sales Private Ltd, National Informatics Centre, Sagem Morpho Securities, Satyam Computer Services, L1 Identity Solutions, Totem International, Linkwel Telesystem, Sai Infosystem India, Geodesic, ID Solutions, NISG, SQTC and Telsima Communication.One contract each was awarded to Aircel, Bharati Airtel, BSNL, Railtel Corporation of India Limited, Reliance Communications and Tata Communications.	
5	NEW DELHI: Chairing a meeting of Union Council of Ministers ahead of the completion of two years of the government on Thursday, Prime Minister Narendra Modi asked his ministerial colleagues to focus on carrying the initiatives taken by the dispensation to the grassroots.Modi, who spoke for over 10 minutes in the meeting, was of the view that there still scope to make people more aware of the initiatives taken by the government.	CHAIN35-["a short film on the two years of the government was also shown to the ministers.Modi" in sentence 1, "its" in sentence 1]

	He said additional efforts, over and above what was being done, were needed to tell the masses of the benefitsgovernment has extended to them.At the meeting, a short film on the two years of the government was also shown to the ministers.Modi also took stock of the Aadhar scheme and the Direct Benefit Transfer scheme and its penetration among the massesand reviewed the functioning of some ministries who had been left out in earlier meetings.He briefly spoke about the various programmes being lined up to mark the government's second anniversary At the meeting, a powerpoint presentation on the visibility of various Union ministries on social media was also made. Modi has been laying emphasis for quite some time on increasing the government's presence on the social media.Ministers have already been asked to be more active on platforms such as Twitter.The Prime Minister has often emphasised that all Union ministers should be aware of initiatives taken by variousministers so that they can answer authoritatively in debates and press conferences.	CHAIN54-["Prime Minister Narendra Modi" in sentence 1, "his" in sentence 1, "Modi" in sentence 2, "the government 's presence on the social media.Ministers" in sentence 2, "the social media.Ministers" in sentence 2, "all Union ministers" in sentence 2, "they" in sentence 2] CHAIN27-["the meeting" in sentence 1, "the meeting" in sentence 1, "the meeting" in sentence 1] CHAIN43-["the government" in sentence 1, "the government 's" in sentence 1, "the government 's" in sentence 2]

Appendix D

Top Three Entities on the basis of Sentiment Score

About the Entity			
Source	**First**	**Second**	**Third**
Hindu	Nandan Nilekani	Manmohan Singh	Nara Chandrababu Naidu
TOI	Narendra Modi	Nandan Nilekani	Sonia Gandhi
HT	Arun Jaitley	Narendra Modi	Jairam Ramesh (Negative)
Indian Express	Narendra Modi	Nandan Nilekani	Rahul Gandhi
Deccan Herald	Nandan Nilekani	Manmohan Singh	Narendra Modi
Telegraph	Manmohan Singh	Nandan Nilekani	Rahul Gandhi
New Indian Express	Arun Jaitley	Narendra Modi	Manohar Gopalkrishna Parrikar

By the Entity			
Source	**First**	**Second**	**Third**
Hindu	Nandan Nilekani	Nara Chandrababu Naidu	P. Chidambaram
TOI	Nandan Nilekani	Narendra Modi	Manohar Gopalkrishna Parrikar
HT	Ajay Bhushan Pandey	Arun Jaitley	Nandan Nilekani
Indian Express	Ravi Shankar Prasad	Sonia Gandhi	Rahul Gandhi
Deccan Herald	Nandan Nilekani	Arun Jaitley	Ravi Shankar Prasad
Telegraph	Nandan Nilekani	Montek Singh Ahluwalia	P. Chidambaram
New Indian Express	Ravi Shankar Prasad	Arun Jaitley	Ajay Bhushan Pandey

References

[1] Sevgi Yigit-Sert, Ismail Sengor Altingovde, and zgr Ulusoy. 2016. Towards detecting media bias by utilizing user comments. In Proceedings of the 8th ACM Conference on Web Science. ACM, 374375.

[2] Carole Spary. 2010. Disrupting rituals of debate in the Indian parliament. The Journal of Legislative Studies 16, 3 (2010), 338351.

[3] Michael Rder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In Proceedings of the eighth ACM international conference on Web search and data mining. ACM, 399408.

[4] Carson Sievert and Kenneth Shirley. 2014. LDAvis: A method for visualizing and interpreting topics. In Proceedings of the workshop on interactive language learning, visualization, and interfaces. 6370.

[5] Kai Sheng Tai, Richard Socher, and Christopher D.Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. CoRR, abs/1503.00075.

[6] Haoyue Shi, Hao Zhou, Jiaze Chen, and Lei Li. 2018. On tree-based neural sentence modeling. CoRR, abs/1808.09644.

[7] Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. 2014. Political ideology detection using recursive neural networks. In proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, volume 1, pages 11131122.

[8] Anirban Sen et al. 2019. Studying the Discourse on Economic Policies in India Using Mass Media, SocialMedia, and the Parliamentary Question Hour Data. ACM SIG-CAS Conference on Computing and Sustainable Societies (COMPASS)(COMPASS 19), July 35, 2019, Accra, Ghana.

[9] David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. Latent dirichlet allocation. Journal of Machine Learning Research, 3:2003, 2003 pages.