Big Data Text Analytics in Financial sector

Mirjana Pejić Bach, Živko Krstić, Sanja Seljan

Citation: Pejić Bach, M., Krstić, Ž. & Seljan, S. (2019) Big data text mining in the financial sector. U: Metawa, N., Elhoseny, M., Hassanien, A. & Hassan, M. (ur.) Expert Systems in Finance: Smart Financial Applications in Big Data Environments. London, Routledge, str. 80-96 doi:10.4324/9780429024061.

Abstract

Big data technologies had impact on different industries in last decade. Financial sector as other sectors around the world concentrated mostly on analysis of structured data but with appearance of big data technologies hidden information from semi-structured and unstructured data could be harvested.

Financial sector and banks are among early adopters of big data technologies with many interesting case studies. In this paper we will present how big data analytics can be used in financial sector with emphasis on analysis of textual data. Goal of text analytics is to listen our customers – their needs, issues, suggestions, opinions. While financial institutions already have some inputs from their customers in some form, benefit of big data technologies is to compare our institution with competitors with same objective metric (usually machine learning generated metric).

This paper will present several data-driven case studies on textual data where big data technology helps those financial institutions to gain new valuable insights:

- Keyword detection
- NER – Name entity recognition
- Gender prediction
- Sentiment analysis
- Topic extraction
- SNA – Social Network analysis

In this paper we will use original data from real financial case studies but the financial institutions will be masked. We will present commonly used case studies adopted by different institutions from financial sector that can be replicated in any institution.

Keywords: big data, text analytics, financial sector, data science

1. Introduction

New big data technologies help institutions from financial sector to differentiate from their competitors. This differentiation is accomplished with better decision-making process due to new insight from big data analysis. Financial sector generates vast amount of data like customer data, logs from their financial products, transaction data and this information can be used in one big data system together with external data like social media data and data from websites (Zha et al., 2013).

This paper will present data-driven case studies from field of text analytics with big data technologies. There are also other interesting case studies in financial sector with big data technologies like risk management, fraud detection, security intelligence with big data.

2. Big Data architecture

2.1. About Big Data

In order to present case studies and their results we need to present foundation for that work. Big data can be described from traditional 3V view (Furht and Villanustre, 2016): Volume, Variety, Velocity. We can also use one more V that is suitable from business view - Value. Volume indicates vast amounts of data coming from different sources (social media, internal systems, networks, logs, sensors) in real time. Our goal is to extract value from this information. Variety indicates diversity of sources and formats coming into our system. We will concentrate on social media data and websites with emphasis on textual data. Velocity indicates speed of data ingestion into our big data system. In our case this mostly depends on scope of our analysis (do we want to track local market and trends or as much as we can collect from entire financial sector). Value as additional V indicate our search for valuable insight in our data that can help us lower costs, increase revenue, optimize processes by analyzing large volumes of data.

There are different versions with even more Vs like Variability, Veracity (Gandomi and Haider, 2015). Even though big data technologies appeared earlier and we had similar technologies before, with big data technologies and distributions we can cope with growing volumes, types and formats that are complex and fast. This is usually when traditional approaches and tools can't help us. We use big data technology to aggregate large amounts of data that come to our big data system from different sources which is important in information extraction and decision making.

2.2. Big Data architectures

We will present two big data architectures that can be used for big data analytics in financial sector: Lambda and Kappa (simplification of Lambda architecture). Other architectures are: Liquid (processing and messaging layers), SMACK (stands for the Apache frameworks: Spark, Mesos, Akka, Cassandra and Kafka). The Lambda architecture (N. Marz and J. Warren, 2015.) is architecture that presents three layers: batch, speed and serving layer.

The batch layer is part of architecture where we can store raw data or processed data as they come into our system from which we can generate views for analysis and presentation purpose. Arbitrary views are stored in serving layer. Batch layer contain all data to recent hour/hours. Because batch layer is time consuming, views stored in serving layer for additional use do not contain fresh and new data (from last hour/hours depending on time of computation of batch layer). New and fresh data is ingested with help from speed layer.

The speed layer is used for real-time computations, transformations and arbitrary views for serving layer and is used as compliment to batch layer. Real-time usually means milliseconds or few seconds of delay. Together with batch and speed layer we have one unique view on our data. Serving layer generates indexes for fast queries on our arbitrary views.

With Lambda architecture we need to maintain two layers (batch and speed) to create one unique view of data (in serving layer) and this problem can be partially addressed with Apache Spark which has both batch and speed layer under one framework. Another option is to use different architecture - Kappa.

Kappa architecture (Jay Kreps, 2014.), has focus on only one layer – speed (stream) layer instead of two, like in Lambda (batch and speed). Here serving layer and its views are created based on one layer and in Lambda we need to unify two layers (batch and speed layer). Each event is processed, transformed, enriched in speed layer as data arrive in our big data system. This is advantage but also complication in cases when we need to deal with duplicates, when we need to cross-reference different real-time events. Serving layer can be any in-memory or persistent database and for text analysis we can even use databases for full-text search.

2.3. Big Data system - example

Usage of these two architecture depends on nature of your analysis. For purpose of these case studies we can use any architecture.

Our hypothetical system will take input from two main source types: internal data and external data (from social media and websites). Goal of our big data system is to collect all relevant textual data about our hypothetical institution and its competitors. These documents will be further enriched and analyzed to extract valuable insights that we can use for different business purposes like customer segmentation, improvements of marketing campaigns, analysis of voice of customers.

Internal data can be transaction data, log data, application data. External data can be from any social media and website. We will select two social media sources: Facebook and Twitter. For website we will select top 10 website about financial news. Each source has its own limitations and unique elements so first step would be to create proper connector for each source. This connector will ingest data from social media to our system thanks to big data technologies. Facebook and Twitter have their own API and they can provide specific information to your system as long as you want to collect data from public pages and groups (Facebook) or public accounts (Twitter). Information like gender of person who wrote comment is not available so we need to create model that will predict this information for us in order to use it in further analysis.

Each connector needs to be built based on some configuration inputs in order to properly get what we need. Since our goal is to get comments related to financial sector we will create list of financial institutions that we will use as configuration parameter for each connector. That list has structure (bank1, bank2, bank3, #bank1, …). Facebook needs also one more input and that is search space (list of pages or groups). In Facebook API we need to send two lists: list of keywords and list of pages or groups. List of keywords is presented before and list of pages and groups has same structure but with emphasis on only public pages and groups (pageid125, pageid554, groupid124, …). There are additional parameters like language filter.

For Twitter we have simpler situation, since for this media we only need list of keywords and Twitter API will return any tweets that contains those keywords in real-time. When we have all parameters entered properly in our big data architecture, our custom data connectors can now start feeding our big data system with real-time or near real-time data. Facebook will bring data in mini-batches and Twitter will be pure real-time feed. Those data sets contain several interesting information like: username, timestamp of comment, page name where comment was posted (Facebook specific), number of likes/favorites on that comment, number of shares/retweets, language detected in that comment, keyword which was used for relevant comment detection.

Websites are more complex. Each website has its own design and layout of comments and articles. Some website is harder, some are simpler and this means that we need to build custom connector (if website has API), web crawl process (if it's possible by terms and conditions) or to use commercial partner that can provide us that information legally. Crawl process is commonly used. For crawling, we need to create crawl connector or process and workflow for each individual website. Parameters would be website domain, crawl start URL (if we want only one category on that website – Financial category), keywords that we want to find. Based on these inputs we need to create specific link extraction process for each website and then specific content crawl process that our system will use to extract articles and comments.

Crawling process will start based on selected parameters and then all relevant URLs will be filtered for content crawl process. When we have full list of relevant URLs (only URLs that are from financial topics) we can crawl all information that we need like: article, author, timestamp of article, all comments of that article, comment author, comment timestamp, etc.

Now we have data sets coming from Facebook, Twitter and top 10 websites from external sources and transnational data, log files from internal sources to get metrics like number of accounts opened, number of transactions, website usage from our website etc. This information will be transformed, cleaned and enriched in our big data environment. Enrichment process is related to our case studies: sentiment analysis, gender prediction.

3. Keyword extraction

With new technologies and analysis in recent times and especially in our case of big data analytics with vast volumes of new data coming from different sources there is a need for keyword extraction. Keyword extraction plays important role in financial sector. It is used in simple form in our previous example when we needed list of keywords to extract related comments and articles from external source. More complex but sophisticated usage would be to use automatic keyword extraction (Hasan and Ng, 2014). This field gain huge interest in past several years since volumes of data are growing and we can't manually read every document or comments. With this combination of keyword extraction and big data we can extract only what we need in very short time with fully automated process eliminating most of manual work and increasing speed of data collection.

Simple example used in presented architecture (list of banks) still help us filter only relevant articles and comments from external sources which helps us bring only data that we need in our further analysis and visualizations. This simple approach can get more detailed if we introduce stemming or lemmatization tasks on keywords in order to gather all variations of specific keywords (example: bank, banking, banks -> bank). This feature is useful in later use and especially for mention counting or online presence metric. This metric practically counts how many mentions of specific keyword we have for specific page name or username. This online presence metric can be used for institution comparison in financial sector (bank1 v bank2). We can see trends in time (bank1 has higher online presence in last month then bank2 and by how much).

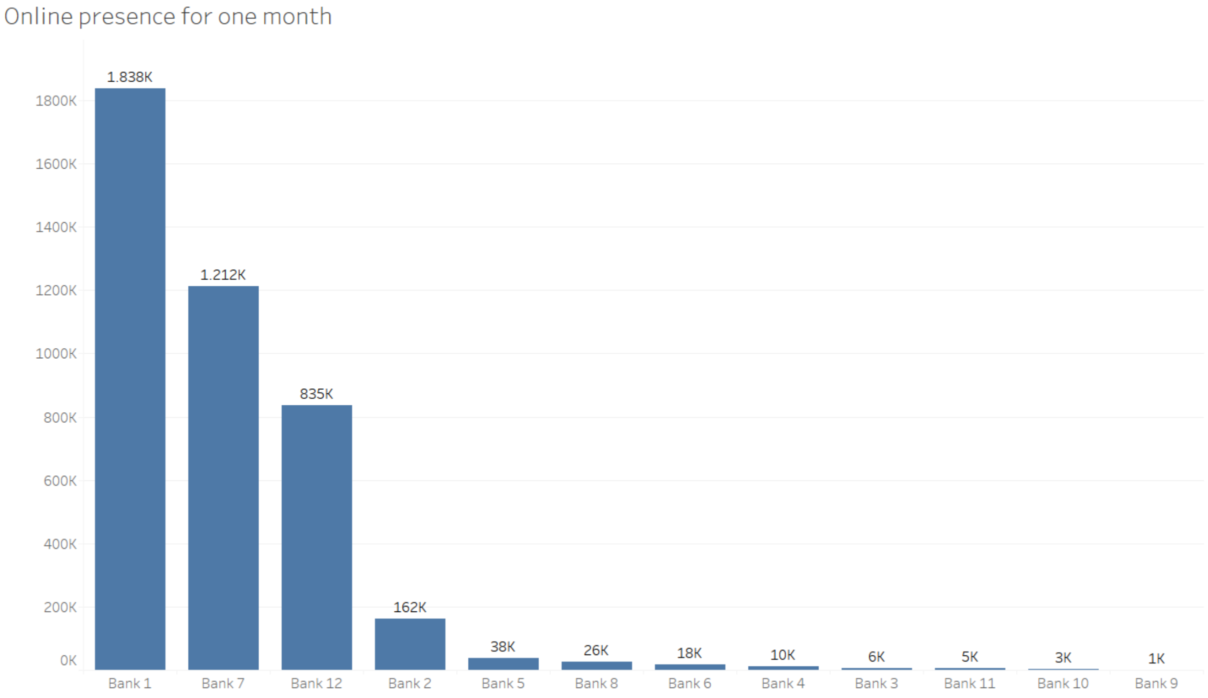


Figure 1. Online presence for 12 banks (data for one month)

On this graph we can see mentions on all sources (Facebook, Twitter and websites) in one month. Bank 1, Bank 7 and Bank 12 have majority of total online presence.
To summarize keyword extraction has 4 approaches (Bharti et al., 2017): Statistical (term frequency, inverse document frequency), Linguistic (WordNet, n-Gram, POS patterns), machine learning (Naïve Bayes) and hybrid approach (some combination of previous three approaches).

4. NER – Named Entity Recognition

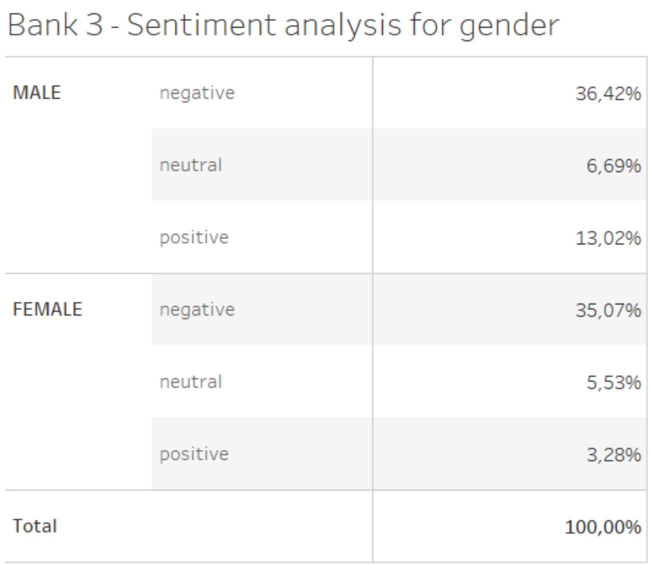
Named entity recognition (Grishman and Sundheim, 1996.) is process which labels some sequence of words in documents which are the names of things (email, amounts-currency, company/bank/institution name, brand name, city-state name, …). Financial institutions use NER for internal data also like extraction of client names, bank account numbers, IBAN number. Example of NER on external data (websites) is presented below:

Shares in the green include Company1 [ORGANIZATION] *as this retailer beat on earnings on an adjusted basis, with revenue topping estimates as well, Company2* [ORGANIZATION] *as the Chicago-based company beat on earnings by a healthy margin, although it missed slightly on revenue, and Company3* [ORGANIZATION] – *shares soaring on the cloud company's public debut, after initialing pricing at $54* [NUMBER] *a share in October of 2017* [TIME]. *A good day for the CEO John Smith* [PERSON].

There are dictionaries with predefined named entities that every organization can use for quick start and result. For better results we need more complex solutions. Since tweets and comments from social media and website usually lack context and are noisy there are more complex solutions like supervised approach for NER (Ritter et al., 2011.).

5. Gender prediction

Information about gender is really useful especially since emphasis of our analysis is on use for marketing and better understanding of customers. Simple approach in solving this problem is to make dictionary of female and male names and then match that dictionary with user names. This can be good approach if we need fast results and some sample of classified gender in order to use that information for some inference. Still since we are analyzing social media and websites there are lot of accounts from different organizations, bots, fake accounts with random names and in that case our approach will only recognize what is in the dictionary.
To solve this limitation next step would be to use NLP models such as bag of word and n-grams or combination of both. This approach analyzes word usage and differences between them and difference between styles. Disadvantage would be again type of our textual data since comments usually contain small amount of words. Features used for this classification task are (Zhang and Zhang, 2010.): words (authors suggest that binary representation is more effective – word exist or not in document), average word or sentence length, POS tags (noun, verb, adjective, adverb), word factor analysis – finding groups of similar work (there are 20 lists – example of conversation list is: know, care, friend, saying). Information gain is used as feature selection and with SVM as classifier accuracy was above 72%.
Latest approach is usage of deep learning techniques to cope with this problem. Accuracy of over 85% was achieved (Bartle and Zheng, 2015) with Windowed Recurrent Convolutional Neural Network. Max pooling was used on sentences to achieve this result.



Bank 3 - Sentiment analysis for gender

MALE	negative	36,42%
	neutral	6,69%
	positive	13,02%
FEMALE	negative	35,07%
	neutral	5,53%
	positive	3,28%
Total		100,00%

Figure 2. Sentiment analysis for one bank by gender

6. Sentiment analysis

Sentiment analysis or opinion analysis is used in financial sector to identify voice of customers. Sentiment analysis (Pang and Lee, 2008.) refers to text analysis or natural language processing techniques which helps us determine writers attitude towards specific topic in this case financial topic. There are several approaches to build accurate sentiment model. Some approaches address to this problem from NLP view other from machine learning view or in current years, more specifically, as deep learning problem. First approach to build sentiment model is to build dictionary of known negative and positive words. For this task we need only extreme polarities and word that can be correctly associated with that polarity. When we have our dictionary then sentiment is calculated by simple count of words found in specific document from our dictionaries. Polarity with more discovered words "Wins" and text is then classified.

Next approach would be to create (large) data set where we have documents that are classified manually (by human) and which we can use for machine learning. Problem can be addressed as classification of two classes (positive or negative) or more (range from 1-5 for sentiment intensity). Features can be unigrams or bigrams or combination of both (Go et al., 2009.). Document term matrix is built based on our features and values in this matrix can be either frequencies like TF or TF-IDF (Term Frequency-Inverse Document frequency) or binary representation. When we are finished with training of ML model on our training data and after correct validation of results we can use this model for classification of new document in our system. In our example of big data architectures, we can use this model on batch data but also in real-time data to perform real-time classification (this depends on your choice of architectures). Accuracy can be greater than 80% even with simple algorithms (Narayanan et al., 2013) with correct feature selection and noise removal process.

Last approach in sentiment analysis would be usage of deep learning techniques (Zhang and Liu, 2018) as specific part of machine learning to solve this problem. In deep learning we have term word embeddings and popular are: word2vec, GloVe. Word embeddings is used to represent words as vectors. With this technique we map similar words to nearby points in continuous vector space. Deep learning is improvement from other approaches and especially in sentiment classification of relatively small documents (tweets, comments) like in our case. Deep learning is even used for tasks that are hard to solve like irony or sarcasm detection (Mandelbaum and Shalev, 2016).

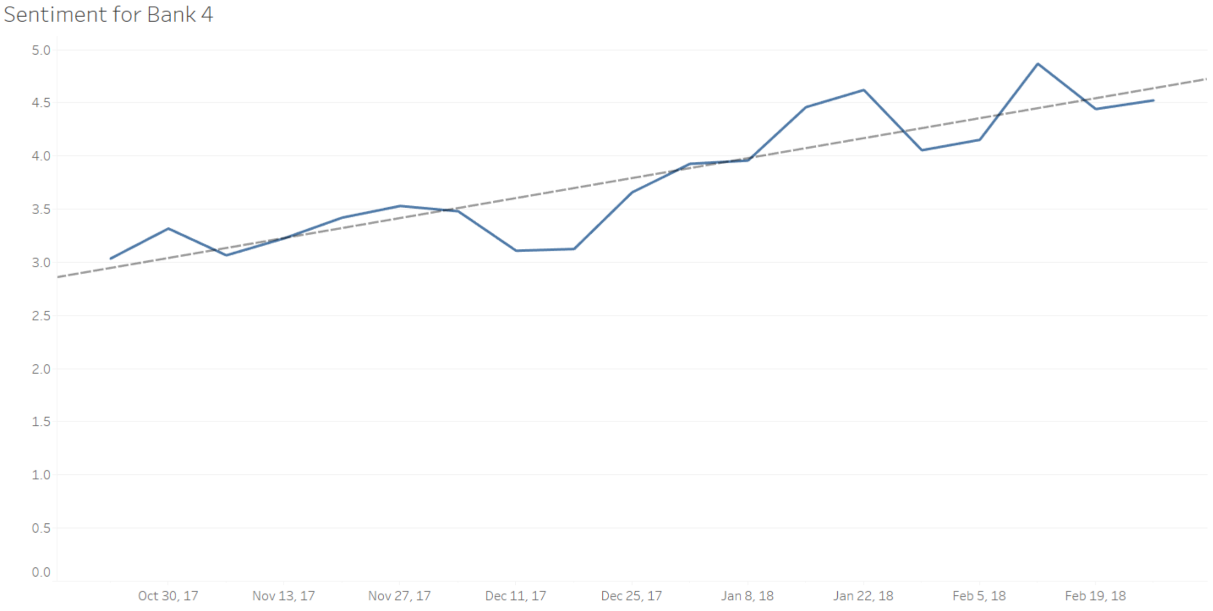


Figure 3. Sentiment analysis over time for one bank

On this graph we can se sentiment analysis over time for one bank. Sentiment values are from 0 to 10 where 0 is bad, 10 is good and 5 would be neutral. We can see that sentiment trend (dotted line) for this bank is rising in selected period (from October 2017 to February 2018).

7. Topic extraction

Data from social media can be used to find discussed topics in certain time period. Papers show that these data can be good source of entity-oriented topics that have low coverage in traditional media news (Zhao et al., 2011). Input in our model would be matrix of document-terms format with TF-IDF frequencies as values or binary representation (0 or 1). Common approach for topic extraction is unsupervised machine learning approach (popular algorithm is LDA). Like in previous example each document that is ingested into our hypothetical system can be assigned to specific topic. Usually topics are time-related. We generate most important topics for specific time period like one week, one month or one year.

New approaches take also deep learning techniques for topic extraction. Popular word embeddings in this case is lda2vec which is modification of word2vec presented in sentiment analysis (Moody, 2016.). Lda2vec uses word2vec principles and expands this to word, document and topic vectors.

Topic extraction helps us answer question “WHAT” is talked about our institution or our competitors. Usually topics are represented as word clouds but they can be visualized by some more complex graphical representation (LDAvis – Intertopic distance map is visualized with PCA).

8. SNA – Social Network analysis

Social network analysis is process that is based on graph theory and used for better understanding of social structures. Since in our hypothetical case we collect data from social media and web we can use those data to describe interactions between users from those sources (example would be Twitter friends and followers). When we talk about SNA we talk about nodes and edges. In our case (let’s take Twitter for example), each node would be one user from Twitter and each edge is relationship between two users (user is connected to other user by follow or retweet). Usual metrics calculated with SNA techniques are (Ediger et al., 2010.):

centrality measures, node degrees (used to find users who are highly connected), closeness (goal is to find users who can spread information to others), clustering coefficient, PageRank.
Social network analysis is different type of analysis in comparison to text analysis but it is used here to show how text analysis and its result can be integrated with this analysis (L'huillier et al., 2011). With this approach we can enrich our previous analysis like topic extraction. One way is to combine SNA graph with topic extraction in order to find out which user in our specific network belongs to topic generated in previous chapter.
Next example of usage of both worlds is when we identify user who can spread message easily on our network of interest (with SNA techniques) then we can use textual data from followers of that user to find out common interests. This information can be used for marketing campaigns to generate best keywords.

9. Conclusion

Big Data text analytics is used by financial sector in many fields to improve customer relationship, marketing campaigns, customer segmentation. This paper analyzed big data architectures and text mining techniques for the Financial sector and the outcome is presented below:

- Customer related data is ingested from popular social media (Facebook and Twitter presented here) and web (popular web pages) with big data technologies. This is used as foundation for text analysis.
- Text analysis is used to extract valuable insights from our collected data sets.
- Most of these techniques can be implemented easily in any financial institution. There are numerous approaches and techniques and finding one depends on needs of financial institution

Future scope

This study can be further extended by covering presented techniques with real data set from specific country. Also, instead of concentrating only on text analysis other case studies and techniques can be presented like fraud detection, churn analysis, risk analysis, security intelligence an else.

References

He, W., Zha, S. and Li, L., 2013. Social media competitive analysis and text mining: A case study in the pizza industry. *International Journal of Information Management*, *33*(3), pp.464-472.

Furht, B. and Villanustre, F., 2016. Introduction to big data. In *Big Data Technologies and Applications* (pp. 3-11). Springer, Cham.

Gandomi, A. and Haider, M., 2015. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*(2), pp.137-144.

Marz, N. and Warren, J., 2015. *Big Data: Principles and best practices of scalable realtime data systems*. Manning Publications Co..

J. Kreps, 2014. "Questioning the lambda architecture", O'Reilly Media, Inc.

Hasan, K.S. and Ng, V., 2014. Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vol. 1, pp. 1262-1273).

Bharti, S.K. and Babu, K.S., 2017. Automatic Keyword Extraction for Text Summarization: A Survey. *arXiv preprint arXiv:1704.03242*.

Grishman, R. and Sundheim, B., 1996. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics* (Vol. 1).

Ritter, A., Clark, S. and Etzioni, O., 2011, July. Named entity recognition in tweets: an experimental study. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 1524-1534). Association for Computational Linguistics.

Zhang, C. and Zhang, P., 2010. Predicting gender from blog posts. *University of Massachussetts Amherst, USA.*

Bartle, A. and Zheng, J., 2015. Gender classification with deep learning.

Pang, B. and Lee, L., 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, *2*(1–2), pp.1-135.

Narayanan, V., Arora, I. and Bhatia, A., 2013, October. Fast and accurate sentiment classification using an enhanced Naive Bayes model. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 194-201). Springer, Berlin, Heidelberg.

Go, A., Bhayani, R. and Huang, L., 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, *1*(12).

Mandelbaum, A. and Shalev, A., 2016. Word embeddings and their use in sentence classification tasks. *arXiv preprint arXiv:1610.08229*.

Zhang, L., Wang, S. and Liu, B., 2018. Deep Learning for Sentiment Analysis: A Survey. *arXiv preprint arXiv:1801.07883*.

Zhao, W.X., Jiang, J., Weng, J., He, J., Lim, E.P., Yan, H. and Li, X., 2011, April. Comparing twitter and traditional media using topic models. In *European Conference on Information Retrieval* (pp. 338-349). Springer, Berlin, Heidelberg.

Moody, C.E., 2016. Mixing dirichlet topic models and word embeddings to make lda2vec. *arXiv preprint arXiv:1605.02019*.

Ediger, D., Jiang, K., Riedy, J., Bader, D.A. and Corley, C., 2010, September. Massive social network analysis: Mining twitter for social good. In *Parallel Processing (ICPP), 2010 39th International Conference on* (pp. 583-593). IEEE.

L'huillier, G., Alvarez, H., Ríos, S.A. and Aguilera, F., 2011. Topic-based social network analysis for virtual communities of interests in the dark web. *ACM SIGKDD Explorations Newsletter*, *12*(2), pp.66-73