Matching Writers to Content Writing Tasks

Authors: Chandrashekhar Bhakuni, Khamir Purohit, Prabir Chakraborty, Ujjval Bhatt, Vikas Sardana (AMIL Nov 19 Group 3)

Abstract

Businesses need content. In various forms and formats and for varied purposes. In fact, the content marketing industry is set to be worth $412.88 billion by the end of 2021. However, according to the Content Marketing Institute, creating engaging content is the #1 challenge that marketers face today. We understand that producing great content requires great writers who understand the business and can weave their message into reader (and search engine) friendly content.

In this project, the team has attempted to bridge the gap between writers and projects by using AI and ML tools. We used NLP techniques to analyze thousands of publicly-available business articles (corpora) to extract various defining factors for each writing sample.

Through this project we aim to automate the highly time-consuming, and often biased task of manually shortlisting the most suitable writer for a given content writing requirement. We believe that a tool like this will have far reaching positive implications for both parties - businesses looking for suitable talent for niche writing jobs as well as experienced writers and Subject Matter Experts (SMEs) wanting to lend their services to content marketing projects. The business gets the content they need, the content writer/ SME gets a chance to leverage his or her talent, while the reader gets authentic content that adds real value.

Overview

We use data-driven techniques to identify some of the many aspects that make a writer *unique*. We then use concepts like Myers–Briggs Type Indicator (MBTI) to extract the traits and abilities of an author, using his/ her write-up.

In order to analyze the writing styles, we started exploring publicly available corpora of long-form content formats like blogs and articles. Based on our subject matter understanding and data scraping feasibility, we zero-ed in on websites like Harvard Business School (https://hbswk.hbs.edu/) and Entrepreneur India (http://entrepreneur.com/) that publish articles in sub-domains like Management, Finance, Strategy, and Leadership.

We built our own dataset using the Beautiful Soup package of Python. Using this method we were able to parse approximately 40,000 articles. Data points such as page URL, article headline, article text, business domain, name of the author were the ones of interest for us.

Textual data, like long-form articles, requires a significant amount of cleaning and pre-process to retain the relevant text. The steps also vary as per the content source. For this project, we cleaned out several unimportant texts. In addition, extra focus was required

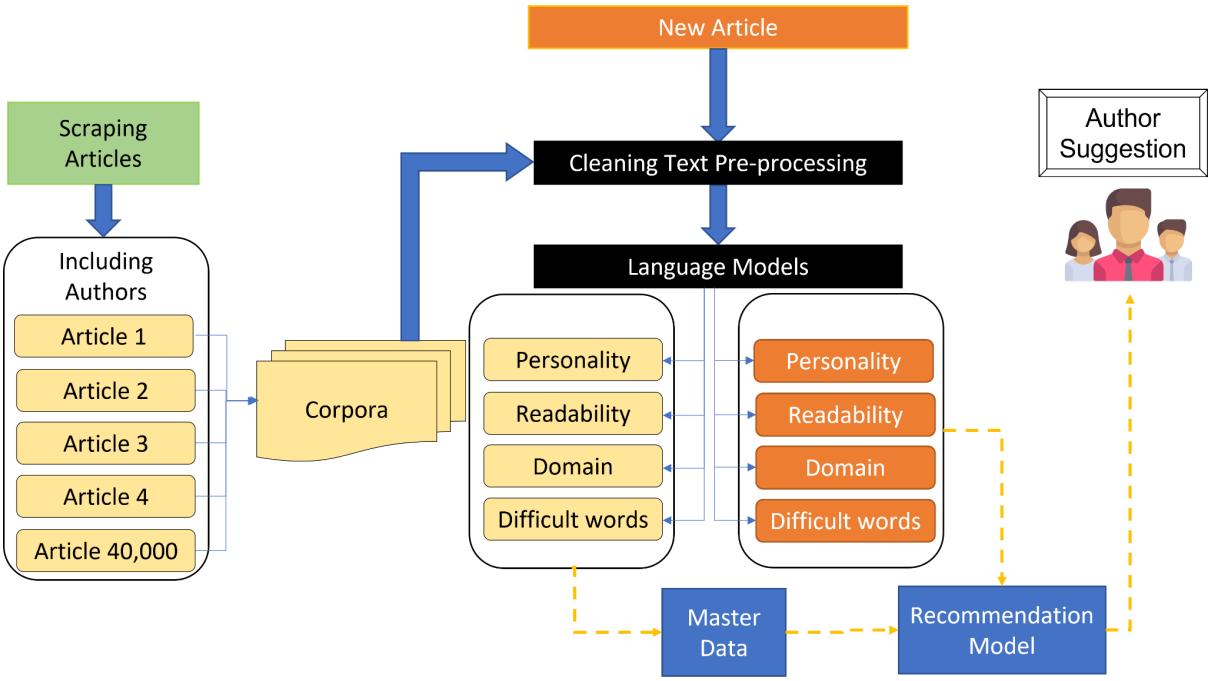
to identify and weed-out special characters and junk elements introduced due to various encoding techniques (like UTF-8, UTF-16 and UTF-32).

In the course of the project, we encountered, explored and implemented various algorithms, methods, and thought processes. The nature of our goal warranted that we work with multiple models in parallel; and eventually combine them to offer a prediction.

We started with multiple NLP libraries and simple ML models like TF-IDF encoder, MultinomialNB classifier, and went on to make complex and multi-layered models that employ Deep Learning, Neural Networks, Universal Sentence Encoder, LSTM with GloVe embeddings, fastText, and its many variations.

In all, the project was a journey of discovery and learning where we developed and experimented with multiple models - either to enhance relevance, or to improve accuracy.

Step-by-step Walkthrough of the Solution



Project Execution Workflow

Build Up To The Final Solution

Steps Taken to Solve the Problem	**Findings at Each Stage**	**Implications for Next Steps**
Scraping data from two sources: HBS and Entrepreneur.com for five domains. Collected approx. 40,000 articles	(a) Different sources had various formats of the articles and different places for author names, etc. (b) Scraping the data using Beautiful Soup library, for different source websites (c) Scraping led to duplicate articles with garbled text	(a) Cleaned out items like static headers and footers, publisher information, list of references, and web page elements like CTA buttons, banners, hyperlink texts, special characters and junk elements.
Cleaning the data using Beautiful Soup and Regular Expressions (Regex) by removing multiple special characters, duplicate records, articles with missing authors or small length (< 50 words), etc.	(a) Dropping articles with duplicate values and missing values. (b) Cleaning of text data required multiple iterations.	(a) Total articles came down to approx. 29,000. (b) With major reduction in the Finance domain.
More than 15 features for articles were calculated using NLP libraries NLTK, TextSTAT and TextBlob.	(a) Flesch_Reading_Ease was chosen for computing the readability. (b) Difficult words with more than two syllables were calculated and an appropriate feature class was created.	The key features were identified for Author recommendation.
To derive MBTI for each article, data (essay corpus) from University of Antwerp was analyzed.	We have used the same MBTI data to train our model on various algorithms. Out of which LSTM gave the best results.	We then used the model to predict the personality traits of the authors in our dataset.
To predict the domain class, we developed multiple models, starting with simple Naive Bayes ML model, to Transfer Learning models, USE, Conv1D + bi directional LSTM, character embedding, hybrid model with sentence and character embedding and GloVe embedding with LSTM.	(a) Embedding only covered partial vocabulary. For example, GloVe only offered 40% of vocabulary of the articles in our dataset. (b) The models were giving lower accuracy and F1 score specifically for Strategy and Leadership domains.	(a) New words and acronyms such as “GDP”, ‘Facebook”, “Tesla”, “Amazon”, etc. were not part of any embeddings. (b) Recommendation for authors for Leadership and Strategy might not be accurate.

Steps Taken to Solve the Problem	Findings at Each Stage	Implications for Next Steps
Combining all the features and making master data for feeding into our recommendation system.	Out of 18+ parameters, we identified the parameters that would be considered for author recommendations.	Domain, Readability, MBTI, and Density of Difficult Words, were the four features selected.
Recommendations using KNN and Cosine Similarity for the selected features by converting the data into One Hot Encoding.	(a) KNN could give effective predictions since the data was sparse. (b) Cosine Similarity was calculated and content based recommendations were derived.	(a) Master data with all the authors was created. It was subsequently used for evaluating the new article.
Features for a new article were derived with (a) Saved models and (b) NLP library. The features were One Hot Encoded and fed to the recommendation system.	We could generate top “n” recommended authors for any new requirement.	

Model Evaluation

As described above, the final model consists of several pieces that help generate the features for existing articles and any new article sample received. Based on these parameters a recommendation system will predict the best match from the existing authors.

1. Domain Classification (NLP Model)
2. Writer Personality (MBTI) Classification (NLP Model)
3. Rule-based Computation
   i. Readability Score: Using Flesch_Reading_Ease score
   ii. Difficult-Word Density

We will describe the models one-by-one:

Domain Classification

The below mentioned algorithms have been used/ explored for finding various aspects of the text articles:

1. Multinomial Naive Bayes (ML Baseline Model) with TF-IDF[1],[2],[3],[4]
2. LSTM with GloVe Embedding[7],[8],[9],[10],[11]
3. Conv1D with Tensorflow Token Embedding[17],[19],[20]
4. Deep Neural Network with Pretrained Token Embedding using Tensorflow Hub’s Universal Sentence Encoder (USE)[5],[6]
5. Conv1D with Character Embedding[18],[13],[14]

6. Bidirectional LSTM with Hybrid Embedding - Pretrained Token Embedding (USE) + Character Embedding
7. GRU[15],[16]
8. BERT/ DistilBERT Tokenizers[12]
9. HuggingFace Transformers

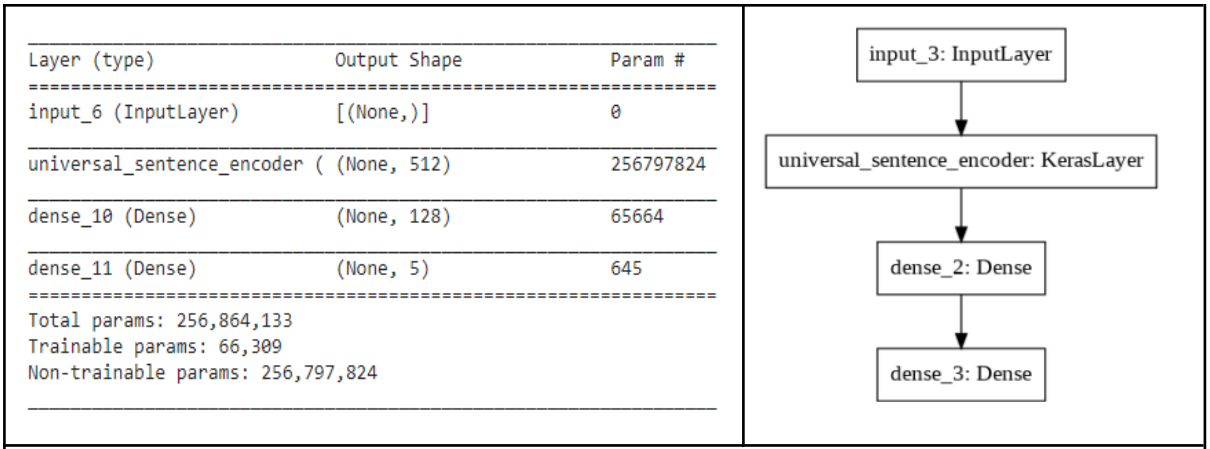
The model was trained on existing data and is able to predict the Domain for a new article. We experimented with multiple embedding techniques, Transfer Learning from pre-trained embeddings and various classification models. We started with a simple Naive Bayes classifier and moved to complex deep neural networks, Bi directional LSTM and Conv1D. We finally picked the best performing model for predicting the domain of a new article.

Accuracy, Precision, Recall and F1-score of various experimental models on test data were tabulated and the model with highest F1-score was picked as the best model.

Model	**Model Description**	**Accuracy %**	**F1-Score**
Baseline: TF-IDF + MultinomialNB	Text vectorization by TF-IDF followed by Naive Bayes Classifier	61.77	0.55
Model 1: Token Embed + Conv1D	Text vectorization followed by word embedding through Keras fed to Conv1D, Global Max and Global Average Pooling layers fed to output dense layer: ● Total params: 3,881,669 ● Trainable params: 3,881,669 ● Non-trainable params: 0	64.44	0.65
Model 2: Pre-trained USE Embedding + Dense	Text vectorization followed by pre-trained Embeddings from Universal Sentence Encoder (USE) fed to the output layer. ● Total params: 256,864,133 ● Trainable params: 66,309 ● Non-trainable params: 256,797,824	73.82	0.73
Model 3: Char Embed + Conv1D	Text vectorization followed by word embedding through Keras fed to Conv1D layer and Global Max fed to output layer. ● Total params: 10,139 ● Trainable params: 10,139 ● Non-trainable params: 0	20.85	0.07

Model 4:	Combination of Character Embedding and USE Embedding fed into BiLSTM and output dense layer. ● Total params: 256,912,243 ● Trainable params: 114,419 ● Non-trainable params: 256,797,824	73.82	0.72
Model 5: GloVe Embed + BiLSTM + Time-distributed	Text vectorization followed by word embedding through GloVe Embedding fed to BiLSTM and Time-distributed and output dense layer. ● Total params: 76,854,457 ● Trainable params: 76,854,457 ● Non-trainable params: 0	69.74	0.69
fastText	fastText model with trigrams and one-vs-all loss was used.		0.72

Both Model 2 and Model 4 gave very close output. However, Model 2 had a slightly higher F1-score and is simpler with only 66K trainable parameters compared to Model 4, which had 114K trainable parameters.

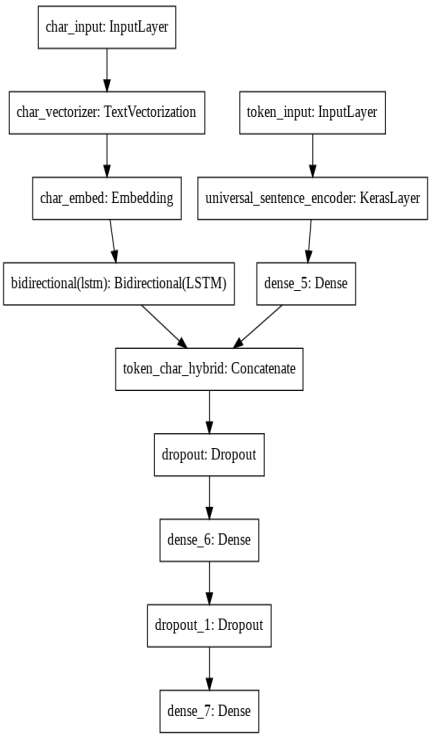


```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_6 (InputLayer)         [(None,)]                 0
_________________________________________________________________
universal_sentence_encoder ( (None, 512)               256797824
_________________________________________________________________
dense_10 (Dense)             (None, 128)               65664
_________________________________________________________________
dense_11 (Dense)             (None, 5)                 645
=================================================================
Total params: 256,864,133
Trainable params: 66,309
Non-trainable params: 256,797,824
_________________________________________________________________
```


Model 2 - Pretrained USE with Dense Layers

Layer (type)	Output Shape	Param #
char_input (InputLayer)	[(None, 1)]	0
token_input (InputLayer)	[(None,)]	0
char_vectorizer (TextVectorizat	(None, 8914)	0
universal_sentence_encoder (Ker	(None, 512)	256797824
char_embed (Embedding)	(None, 8914, 25)	1750
dense_18 (Dense)	(None, 128)	65664
bidirectional_1 (Bidirectional)	(None, 50)	10200
token_char_hybrid (Concatenate)	(None, 178)	0
dropout_3 (Dropout)	(None, 178)	0
dense_19 (Dense)	(None, 200)	35800
dropout_4 (Dropout)	(None, 200)	0
dense_20 (Dense)	(None, 5)	1005

Total params: 256,912,243
Trainable params: 114,419
Non-trainable params: 256,797,824



Model 4 - Pre-trained USE Embedding with Char Embed and Bidirectional LSTM

Both these models use 256M+ non-trainable parameters from TFHub's Universal Sentence Encoder. We do not expect much variation and either of these two models can be used for prediction without much deviation in accuracy (74%). There is a chance that some new articles might be classified incorrectly.

Writer Personality using MBTI

The objective of MBTI[20],[21],[22],[23],[24],[25],[26],[27],[28],[29],[30] classification was to predict the personality of the writer based on the article the writer has written. We have trained our model using the MBTI data consisting of 8675 rows, wherein each article is tagged by its personality trait.

We have deployed different models using algorithms like Naive Bayes, Random Forest, XGBoost, Stochastic Gradient Descent, Logistic Regression, KNN, SVM and finally LSTM. Out of all the models we got the best accuracy (85%) using LSTM.

Readability

Flesch Reading-Ease Scores (FRES) were considered for evaluating the readability of the articles. The formula for FRES is:

$$206.835 - 1.015\left(\frac{\text{total words}}{\text{total sentences}}\right) - 84.6\left(\frac{\text{total syllables}}{\text{total words}}\right)$$

The result is a number that corresponds with a U.S. grade level (Grade 5-10, College, Grad College, and Professional), which denotes an increasing level of complexity.

Difficult-Word Density

Out of total words of the article, words with two or more syllables were defined as difficult words. Accordingly, four classes were created, viz., Basic, Elementary, Intermediate, and Advanced.

```
recommendations('Predicted Author')

[4761, 9570, 4760, 4759]

['Mark Abell', 'Roberta Holland', 'Eyal Shinar', 'Sona Jepsen']
```

```
X1[(X1.shape[0]-1):X1.shape[0]]
```

Cleaned_Author	ESFP	INFJ	INFP	INTJ	INTP	College	Grade_10	Grade_5	Grade_6	Grade_7	Grade_8	Graduate	Professional	Finance	Leadership	Marketing	Strategy	Technology	Advanced	Basic	Elementary	Intermediate
Predicted Author	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0

```
X1[4761:4762]
```

Cleaned_Author	ESFP	INFJ	INFP	INTJ	INTP	College	Grade_10	Grade_5	Grade_6	Grade_7	Grade_8	Graduate	Professional	Finance	Leadership	Marketing	Strategy	Technology	Advanced	Basic	Elementary	Intermediate
Mark Abell	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0

Recommendation of Most Suitable Authors

The articles were categorized in different classes of Readability, Domain, MBTI and Difficult-Word Density. Thereafter the different classes were encoded using OHE. Similarly, the sample text was categorized using OHE.

The Similarity Score using Cosine similarity was calculated for OHE matrices of corpora and the sample text. Based on the maximum cosine similarity value, the author(s) written the most similar text as compared with the sample text were recommended.

Visualization(s)

Include all relevant visualizations that support the ideas/ insights that you gleaned from the data. Scraped data from websites were cleaned up.

Article URL	Headline	Article Text	Sub Domain	Domain	Author
https://www.entrepreneur.com/article/370874	\n\t\t\t\tIs Launching a New Brand the Right M...	\n\n\n\t\t\t\t\t\t\t\tJune\n\t\t\t\t\t\t\t\t10...	Branding	Marketing	Melissa Packham
https://www.entrepreneur.com/article/372254	\n\t\t\t\t3 Reasons Simple Isn't Always Better...	\n\n\n\t\t\t\t\t\t\t\tJune\n\t\t\t\t\t\t\t\t10...	Branding	Marketing	Zaheer Dodhia
https://www.entrepreneur.com/article/368356	\n\t\t\t\tFrom Idea to Revenue: A Six-Step For...	\n\n\n\t\t\t\t\t\t\t\tJune\n\t\t\t\t\t\t\t\t10...	Launching a Business	Marketing	Jessica O'Connell
https://www.entrepreneur.com/article/373825	\n\t\t\t\tAvoid This Common Mistake When Writi...	\n\n\n\t\t\t\t\t\t\t\tJune\n\t\t\t\t\t\t\t\t10...	Writing a Book	Marketing	R. Paulo Delgado

Cleaned and Headline Article Text, with calculated Article Size, etc.

	Domain	Cleaned_Article	Cleaned_Headline	Cleaned_Author	Word_Count	Article_Size
0	Marketing	One of the most common challenges I see client...	Is Launching a New Brand the Right Move for Yo...	Melissa Packham	1013	Large
1	Marketing	Lets say that, like me, youve been thinking ab...	3 Reasons Simple Isn't Always Better When It C...	Zaheer Dodhia	882	Medium
2	Marketing	If youre like most entrepreneurs, you may have...	From Idea to Revenue A Six-Step Formula to Lau...	Jessica O'Connell	1409	Large
3	Marketing	I had an interesting project come across my de...	Avoid This Common Mistake When Writing Your Fi...	R. Paulo Delgado	952	Medium
4	Marketing	In a world where personal touch matters, why a...	Is it Time to Say RIP to RFPs	Heather Ripley	723	Medium

We have 29,631 articles the article distribution is:

Size-Wise

Small (<500)	6164
Medium (501 - 1000)	16576
Large (>1000)	6891

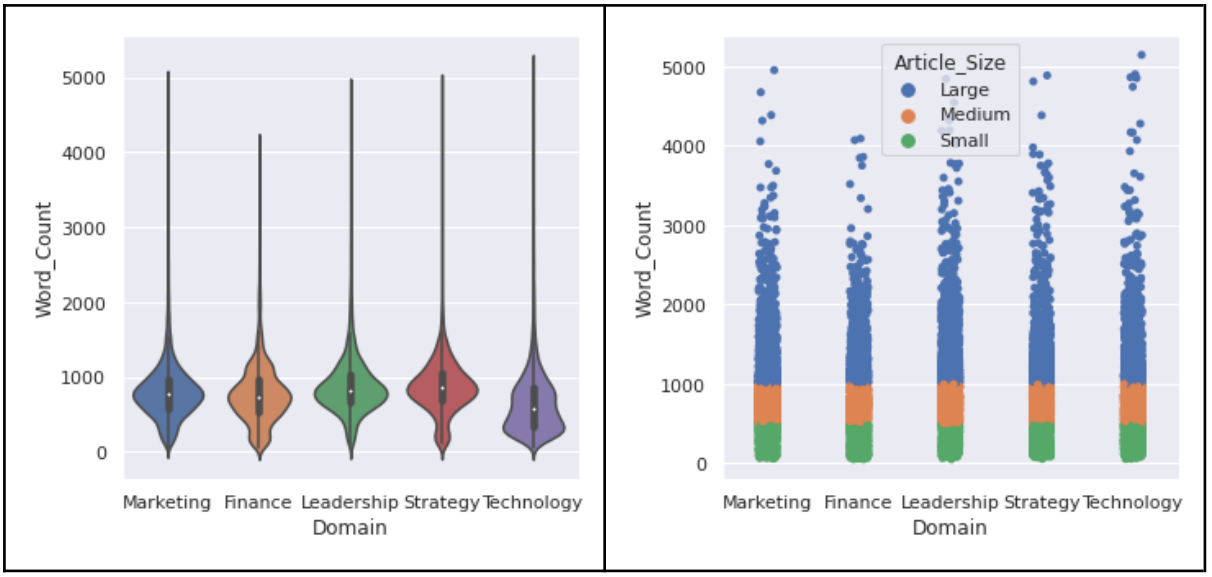
Domain-Wise

Finance	4600
Leadership	6267
Marketing	7467
Strategy	5205
Technology	6092

Cross Distribution

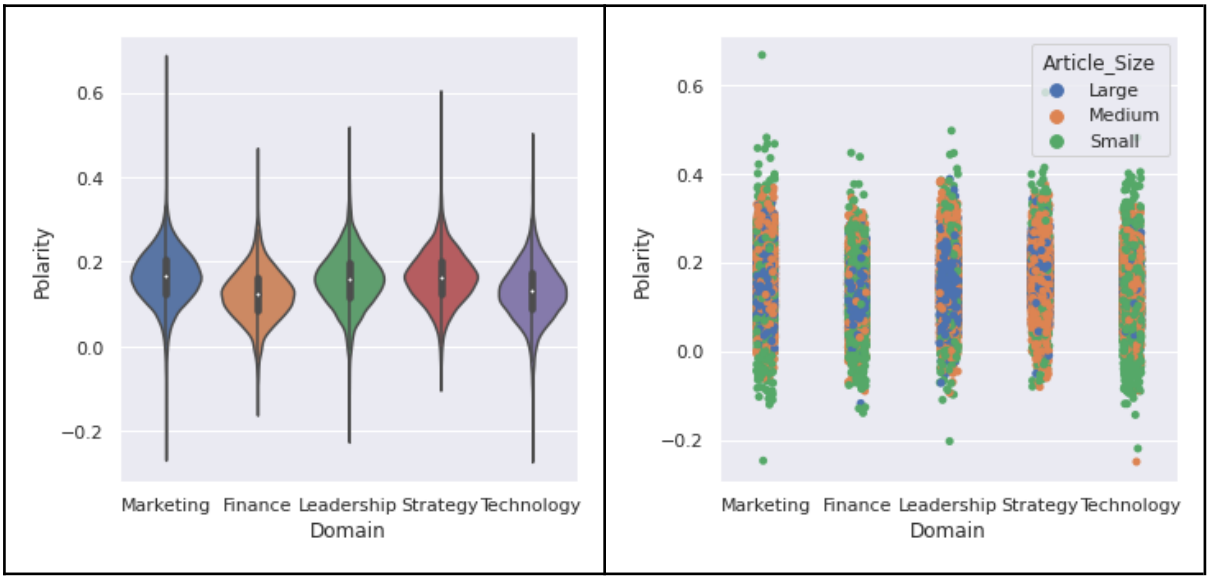
	Large	**Medium**	**Small**
Finance	1031	2553	1016
Leadership	1736	3889	642
Marketing	1632	4572	1263
Strategy	1578	3007	620
Technology	914	2555	2623

We have the highest number of articles for Marketing, and the least number of articles for Finance. We also observe that most articles belong to the Medium-size category.



Sentiment-wise Distribution:

		Domain				
		Finance	**Leadership**	**Marketing**	**Strategy**	**Technology**
Sentiment	**Negative**	126	77	85	44	158
	Neutral	1470	1013	934	698	1740
	Positive	3004	5177	6448	4463	4194

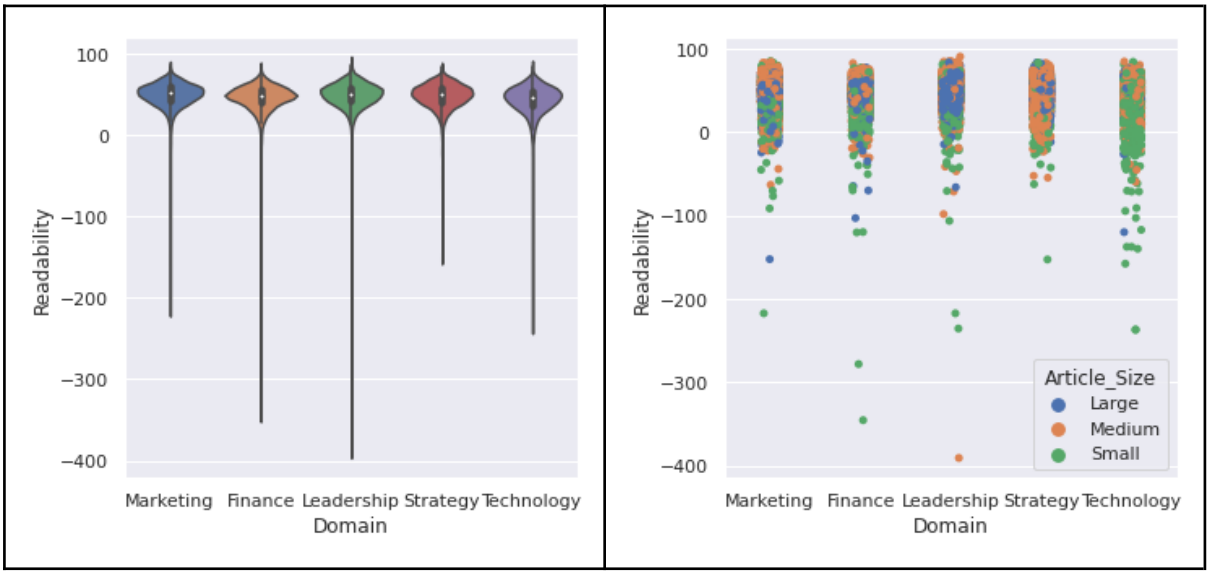


After the sentiment analysis we observe that Positive articles have the highest percentage followed by Neutral and Negative.

Flesch_Reading_Ease:

F_Readability	Professional	Graduate	College	Grade 10	Grade 8	Grade 7	Grade 6	Grade 5
Finance	97	447	2089	1312	572	82	1	0
Leadership	71	425	2495	1882	1075	291	27	1
Marketing	106	450	2928	2303	1341	325	14	0
Strategy	77	432	2124	1491	888	183	10	0
Technology	241	781	2738	1517	706	104	5	0

Most of the articles fall into College level reading ease followed by Grade 10. Also, we have almost zero percentage of Grade 5 level.



Business Impact

We believe that our solution is a good attempt to achieve the intended business outcome. The business content writing space is a fast evolving and dynamic field that is attracting multiple authors and content professionals - which will, in turn, only complicate the problem of identifying the best writers for a given business requirement. The use of AI and ML techniques can bring scale and relevance, while improving speed, accuracy, and depth and eliminate bias due to human errors and limitations.

Recommendations

We strongly believe that this solution has the potential to eliminate manual project allocation processes. Content creators will be able to create their profiles on various creator platforms and upload their entire portfolios for machine-based evaluation and cataloging without worrying about breach of intellectual property or infringement of creative licences.

We recommend using (a further refined version of) our solution and building a platform that can be leveraged by all types of businesses including creative agencies, consulting companies, content platforms, freelancing platforms, etc. This will open up multiple opportunities by bridging the gap between content creators and content seekers and democratize the content space that is currently run by closed groups with major barriers to entry. It can help companies drive down content creation costs and while dramatically shooting up content recency and relevancy.

Limitations of the Solution

Just like any other machine learning solution, our solution is only as good as the data we have used to train it. For this project, we have sourced data from only two portals that usually do not marry branded content. This means our data could be biased (e.g. based on the editorial guidelines and the demographic target audience that those portals cater to), and not a true representation of the different types of writing styles that clients in the real-world might be seeking.

Our solution has been trained to predict writers for just a handful of business domains detailed above. While this was a conscious decision for the scope of this project, this is also a limiting factor as our model has not been exposed to diverse data. This can lead to biased predictions. Moreover, as described above, while our training data is multifaceted, it is also imbalanced.

Modern-day business content writing requirements are not limited to long-form articles and blogs (like the ones we have used for our project). Content marketing strategies demand various other forms of content writing (and copywriting) like social media posts, info-graphics, research reports, whitepapers, technical case studies, PoVs, business plans, video scripts, and more. At this stage, our solution is not capable of identifying writers for these varied content formats.

Our solution also assumes that clients (content seekers) will have a sample write-up for the kind of output they are seeking; and this will be considered as the input for our production. However, this might not be true in the real world. We understand that clients rarely have reference write-ups. Instead, they have creative briefs and RFPs; or they simply prefer to talk to a writer and share a brief with him or her. They also prefer to have a conversation to explain their requirements.

Ideas to Enhance the Solution

The solution can be improved in multiple ways. Some of these could have a significant amount of improvement, while some others are simple enhancements that can have an incremental effect in the accuracy, applicability, and relevance. Here are the top four from our wishlist:

1. **More Data:** While no amount of data can be termed as “enough data”, we believe our solution can be far more relevant and real-world-ready with diverse data with more classes, writing styles and formats. Data from various sources across geography, demography, domains (content and vocabulary for specific domain), and tone of voice should be used. Special focus on portals that carry “branded” or “sponsored” content would bring in a lot more variety and hence a lot more real-life learning for our models.

2. **Different Models:** Use of different pretrained embeddings and transfer learning from pretrained models can be further explored to improve the domain classification results. This can include various BERT based pretrained models, DistilBERT, ALBERTA, etc. Fine Tuning pretrained transformer based text classification models from HuggingFace 🤗 can also help improve the domain classification models.

3. **More Parameters:** The quest for finding the “best writer” is not an easy one. This is because there is no real definition for good writing. While this project uses some of the most basic parameters, a real-world writer needs to be examined on various other parameters. In the world of digital content, search engine optimization, and action-oriented writing, a good writer is expected to have mastered a lot more parameters like Grammar, Subject Matter, Formatting, Consistency, Structure, Call to Action, Product Placement, Depth and Recency of Research, Redundancy, Hyperlinks (and interlinks), use of Search Keywords, SEO Best Practices, Originality (plagiarism-free), and Adherence to the brand's Tone of Voice and Style Guide.

4. **More Computation Power:** Our models use Deep learning consisting of neural networks with multiple hidden layers. In addition, we are also using unsupervised learning algorithms for obtaining vector representations for words. These correspond to particularly demanding needs in terms of computational resources. As we plan to use more and more data, with a higher number of features and parameters, we believe that we will be requiring high-performance computational resources that can support multiple epochs and the corresponding long training times.

Key Learnings from the Process

The process gave us a great opportunity to plan and implement the entire project lifecycle - from problem identification and data collation to implementation and execution. As students, it was also an ideal setting to identify our interests, build on our strengths and work on our weaknesses.

The team has been learning and implementing new ideas and exploring different approaches throughout the project duration. We have also realized that several machine learning libraries, DL methods, NLP approaches, are dynamic and one needs to learn continuously. We learned that knowledge is the key to success and good data is divine.

Note: The project was fully conceptualized, coded, and delivered remotely, and in the backdrop of a pandemic, it allowed us to develop useful skills for remote collaboration, communication, and teamwork.

Ideas for Future Projects

The confluence of machine learning and human languages is an interesting and extremely dynamic space. In order to stay relevant, we will have to stay abreast with the developments in the field and keep innovating. For our next project, we believe we will have a lot more maturity. We would like to spend more time with data identification and data collation.

References

[1] Michael J.Garbade. “ A Simple Introduction to Natural Language Processing” .Available at https://becominghuman.ai/a-simple-introduction-to-natural-language-processing-ea66a1747b32

[2] McCallum, Andrew. “Graphical Models : Bayesian Network Representation. Available at https://people.cs.umass.edu/~mccallum/courses/gm2011/02-bn-rep.pdf

[3] Hastie, Trevor. (2001). The elements of statistical learning : data mining, inference, and prediction : with 200 full-color illustrations. Tibshirani, Robert., Friedman, J. H. (Jerome H.). New York: Springer. ISBN 0-387-95284-5. OCLC 4680922 Available at https://en.wikipedia.org/wiki/Special:BookSources/0-387-95284-5

[4] Zhang, Harry. The Optimality of Naive Bayes Available at http://www.cs.unb.ca/~hzhang/publications/FLAIRS04ZhangH.pdf

[5] Auer, Peter; Harald Burgsteiner; Wolfgang Maass (2008). "A learning rule for very simple universal approximators consisting of a single layer of perceptrons" (PDF). Neural Networks. 21 (5): 786–795. Available at https://web.archive.org/web/20110706095227/http://www.igi.tugraz.at/harry/psfiles/biopdelta-07.pdf

[6] Schmidhuber, Jürgen (2015-01-01). "Deep learning in neural networks: An overview". Neural Networks. 61: 85–117 Available at https://www.sciencedirect.com/science/article/abs/pii/S0893608014002135?via%3Dihub

[7] Ralf C. Staudemeyer, Eric Rothstein Morris (12 Sep 2019. “Understanding LSTM -- a tutorial into Long Short-Term Memory Recurrent Neural Networks” Available at - https://arxiv.org/pdf/1909.09586.pdf

[8] Sak, Hasim; Senior, Andrew; Beaufays, Francoise (2014). "Long Short-Term Memory recurrent neural network architectures for large scale acoustic modeling" Available at https://web.archive.org/web/20180424203806/https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/43905.pdf

[9] Li, Xiangang; Wu, Xihong (2014-10-15). "Constructing Long Short-Term Memory based Deep Recurrent Neural Networks for Large Vocabulary Speech Recognition". Available at arXiv:1410.4281

[10] Felix A. Gers; Jürgen Schmidhuber; Fred Cummins (2000). "Learning to Forget: Continual Prediction with LSTM". Neural Computation.Available at https://en.wikipedia.org/wiki/Neural_Computation_(journal)

[11] Vaswani, Ashish & Shazeer, Noam & Parmar, Niki & Uszkoreit, Jakob & Jones, Llion & Gomez, Aidan & Kaiser, Lukasz & Polosukhin, Illia. (2017). Attention Is All You Need. Available at - arXiv:1706.03762v5

[12] Devlin, Jacob & Chang, Ming-Wei & Lee, Kenton & Toutanova, Kristina. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Available at - arXiv:1810.04805v2

[13] Dor Bank, Noam Koenigstein, Raja Giryes (12 March 2020). “Autoencoders” Available at - arXiv:2003.05991v1

[14] Cho, Kyunghyun; van Merrienboer, Bart; Gulcehre, Caglar; Bahdanau, Dzmitry; Bougares, Fethi; Schwenk, Holger; Bengio, Yoshua (2014). "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation".Available at - arXiv:1406.1078

[15] Denny Britz "Recurrent Neural Network Tutorial, Part 4 – Implementing a GRU/LSTM RNN with Python and Theano – WildML". Available at http://www.wildml.com/2015/10/recurrent-neural-network-tutorial-part-4-implementing-a-grulstm-rnn-with-python-and-theano/

[16] Gruber, N.; Jockisch, A. (2020), "Are GRU cells more specific and LSTM cells more sensitive in motive classification of text?", Frontiers in Artificial Intelligence . Available at https://www.frontiersin.org/articles/10.3389/frai.2020.00040/full

[17] Fen Li,Ming Liu,Yuejin Zhao,Lingpin kong “Feature extraction and classification of heart sound using 1D convolutional neural networks” Available at https://asp-eurasipjournals.springeropen.com/articles/10.1186/s13634-019-0651-3

[18] Chih-Cheng Chen, Zhen Liu,Guangsong Yang,Chia-Chun Wu,Qiubo Ye “An improved Fault Diagnosis using 1D-Convolutional Neural Network Model Available at https://www.mdpi.com/2079-9292/10/1/59/pdf

[19] Badi, Martín; Barham, Paul; Chen, Jianmin; Chen, Zhifeng; Davis, Andy; Dean, Jeffrey; Devin, Matthieu; Ghemawat, Sanjay; Irving, Geoffrey; Isard, Michael; Kudlur, Manjunath; Levenberg, Josh; Monga, Rajat; Moore, Sherry; Murray, Derek G.; Steiner, Benoit; Tucker, Paul; Vasudevan, Vijay; Warden, Pete; Wicke, Martin; Yu, Yuan; Zheng, Xiaoqiang (2016). "TensorFlow: A System for Large-Scale Machine Learning" (PDF). Available at arXiv:1605.08695

[20] Dean, Jeff; Monga, Rajat; et al. (November 9, 2015). "TensorFlow: Large-scale machine learning on heterogeneous systems" Available at http://download.tensorflow.org/paper/whitepaper2015.pdf

[21] G. Farnadi, G. Sitaraman, S. Sushmita, F. Celli, M. Kosinski, D. Stillwell, S. Davalos, M. Moens and M. D. Cock, Computational personality recognition in social media, User Modeling and User-Adapted Interaction, vol.26, nos.2-3, pp.109-142, 2016. [Online] available at https://www.researchgate.net/publication/293194512_Computational_personality_recognition_in_social_media

[22] T. Ryan and S. Xenos, Who uses Facebook? An investigation into the relationship between the Big Five, shyness, narcissism, loneliness, and Facebook usage, Computers in Human Behavior, vol.27, no.5, pp.1658-1664, 2011.[Online] available at https://isiarticles.com/bundles/Article/pre/pdf/32237.pdf

[23] C. J. Soto and J. J. Jackson, Five-factor model of personality, in Oxford Bibliographies in Psychology, D. S. Dunn (ed.), Oxford University Press, NY, 2013. [Online] available at https://www.researchgate.net/publication/264476432_Five-Factor_Model_of_Personality

[24] R. R. McCrae and P. T. Jr. Costa, The five-factor theory of personality, in Handbook of Personality: Theory and Research, O. P. John, R. W. Robins and L. A. Pervin (eds.), Guilford Press, NY, 2008. [Online] available at https://www.researchgate.net/publication/264476432_Five-Factor_Model_of_Personality

[25] Ateş, U. (2014). Inference of Personality Using Social Media Profiles, (June). [Online] available at http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.685.8347&rep=rep1&type=pdf

[26] Obinna Harrison Ejimogu (2017) Predicting Personality From Facebook Data: A Neural Network Approach, [Online] available on http://docs.neu.edu.tr/library/6688309210.pdf

[27] P. T. Costa and R. R. McCrae, Revised NEO Personality Inventory (NEO-PI-R) and NEO FiveFactor Inventory (NEO-FFI), 1992. [Online] available on https://www.researchgate.net/publication/285086638_The_revised_NEO_personality_inventory_NEO-PI-R

[28] O. P. John and S. Srivastava, The Big Five trait taxonomy: History, measurement, and theoretical perspectives, in Handbook of Personality: Theory and Research, Guilford Press, NY, 1999. [Online] available on https://darkwing.uoregon.edu/~sanjay/pubs/bigfive.pdf

[29] Asadzadeh Laleh; Rahimi Shahram (2017) Analyzing Facebook Activities for Personality Recognition, [Online] available on 10.1109/ICMLA.2017.00-29

[30] Obinna Harrison Ejimogu (2017) Predicting Personality From Facebook Data: A Neural Network Approach, [Online] available on http://docs.neu.edu.tr/library/6688309210.pdf