Coding for Culture, Diversity, Gender, and Identity: the Potential for Automation in Research

Ms. Chloe Wiggins, Designing Education Lab

Chloe Wiggins is a graduate of Stanford University who majored in Civil Engineering with a concentration in structures and construction.

Dr. Sheri Sheppard, Stanford University

Sheri D. Sheppard, Ph.D., P.E., is professor of Mechanical Engineering at Stanford University. Besides teaching both undergraduate and graduate design and education related classes at Stanford University, she conducts research on engineering education and work-practices, and applied finite element analysis. From 1999-2008 she served as a Senior Scholar at the Carnegie Foundation for the Advancement of Teaching, leading the Foundation's engineering study (as reported in Educating Engineers: Designing for the Future of the Field). In addition, in 2011 Dr. Sheppard was named as co-PI of a national NSF innovation center (Epicenter), and leads an NSF program at Stanford on summer research experiences for high school teachers. Her industry experiences includes engineering positions at Detroit's "Big Three:" Ford Motor Company, General Motors Corporation, and Chrysler Corporation.

At Stanford she has served a chair of the faculty senate, and recently served as Associate Vice Provost for Graduate Education.

Dr. Shannon Katherine Gilmartin, SKG Analysis

Shannon K. Gilmartin, Ph.D., is a Senior Research Scholar at the Michelle R. Clayman Institute for Gender Research and Adjunct Professor in Mechanical Engineering at Stanford University. She is also Managing Director of SKG Analysis, a research consulting firm. Her expertise and interests focus on education and workforce development in engineering and science fields.

Mr. Benedikt von Unold, Stanford University

Benedikt studied Medical Engineering and Mechanical Engineering at the Technical University of Munich (TUM). In 2017, he joined the Designing Education Lab at Stanford University to learn more about the integration of user backgrounds in design. He was involved in various entrepreneurial activities and worked as a student in small, medium and large companies. The creation of innovation was both an essential part in his studies as it was in his jobs.

Dr. Tua A. Björklund, Aalto University Design Factory

Tua Björklund is one of the co-founders and the head of research at Aalto University Design Factory. She conducts and leads research, teaches product design, and facilitates pedagogical development at the Design Factory. Tua has a DSc degree in industrial engineering and management and a MA degree in cognitive science.

Michael Arruza Cruz

1 Introduction and Background

Starting in the summer of 2015, a team at Stanford has been working on designing, revising and offering an engineering course focused on who are today's engineers, and how those engineers consider the people they are engineering for. The course, called Expanding Engineering Limits (EEL): Culture, Diversity and Gender, was first offered in the Fall of 2015, and in revised forms in Winter 2017 and Winter 2018. The learning objectives for students in the course (as of the Winter 2018 offering) are the following:

(1) Identify and analyze the interdependencies of diversity, culture, and engineering, using a variety of research-based sources.
(2) Connect issues relating to diversity and culture to students' experiences in college and future workplace experiences.
(3) Envision new engineering processes, practices, and cultures that reflect expanded perspectives on gender, diversity, and intersectional identities.

In order to better understand the role(s) of such a course in an engineering student's education and how engineering education considers these issues, the instructor team invited two undergraduate researchers to undertake projects in support of these goals. One of these students (Amber Levine) was tasked with identifying other courses across the U.S. with similar subject matter and learning objectives ("EEL Related Courses Study"); she found 13 courses across twelve institutions that connected issues of diversity and culture to engineering and were targeted to engineering students (Levine, 2016). The other student (Chloe Wiggins, who is the lead author on this paper) was tasked with identifying how often the major themes of the course were covered in key engineering education journals ("EEL Terminology Study"); her summer work became the stimulus for the current paper, and her summer findings are summarized in Section 2.1.

Chloe's initial categorization of relevant themes in engineering education journals was done "by hand", in a non-automated way. One can argue that this type of periodic tracking/checking is one means/component of assessing coverage of topics over time. We note that a recent JEE article (Pawley, Schimpf, & Nelson, 2016) implemented content analysis methods to investigate how gender is covered in JEE, and as such is an important indicator paper (Pawley et al.'s work focused on gender, and does not include a broader range of terms relating to culture and diversity). Increasingly, however, word-search algorithms are being used as another way to monitor and measure written language. For example, such algorithms track the use of gendered pronouns in internet-based text (http://genderedinnovations.stanford.edu/case-studies/nlp.html#tabs-3) and are leading to new thinking of "debiasing" language. Simple word clouds are another means of depicting trends. These methods are appealing because of the speed at which coding can be accomplished and the assurance that results will be consistent and accurate.

This paper is designed to examine the development and application of one such algorithm, that we call the "Word Embedding Based Document Ranking Algorithm" (the Document Ranking Algorithm, for short). A student (Michael Arruza Cruz) in the 2nd offering of EEL saw how Chloe's categorization might be automated. His algorithm is described in Section 2.2. The rankings produced by the Document Ranking Algorithm are then compared against manual coding in three types of data on engineering: the journal articles that inspired the development of the algorithm (case 1), engineering project document data (case 2) and interview transcripts on engineers workplace experiences (case 3). Our conclusions reflect on the potential and limits of automation for coding.

2 Methodology

2.1 Process for Coding by Hand

The population for the EEL Terminology Study includes articles published in three peer-reviewed and highly focused journals on engineering and engineering education journals from 1996 to 2016: *International Journal of Engineering Education* (*IJEE*), *Journal of Engineering Education* (*JEE*), and *Journal of Women and Minorities in Science and Engineering* (*JWMSE*). To begin this investigation, this paper considers publications in these journals during the last ten years, that used the words “culture,” “diversity,” “gender,” or “identity” as themes in their research questions, keywords, and/or findings. Papers published in these journals were considered appropriate for this study if they used at least one of these words three times or more. Note that this investigation did not include papers using related words (e.g. “race,” “ethnicity,” “female,” “femininity,” etc.). Based on this approach, we identified and considered 118 papers in the JEE sample and 104 papers in the IJEE sample. Since JWMSE’s audience is broader, this study focuses only on the articles related specifically to engineering (including engineering courses and introductory science and mathematics courses typically included in required curriculum for engineering students) to make the samples more comparable. This approach yielded 118 papers in the JWMSE sample. A single article could be flagged for multiple words. Many papers didn't explicitly define their terms. Therefore, we used context clues and the themes to come up with a definition for each term.

Table 1 shows the distributions of papers among keywords and publications. It shows for each publication what percentage of its flagged papers were about each keyword. The total percentage for each publication is over 100% because a paper could be flagged for multiple keywords. There were significant areas of overlap depending on the publication.

Table 1: Papers in Journals, 2006-2016, Handcoding

Keyword	JEE (total papers 2006-2016)	Percent with Keyword in JEE	IJEE (total papers 2006-2016)	Percent with Keyword in IJEE	JWMSE (total papers 2006-2016)	Percent with Keyword in JWMSE
Culture	52	34.7%	39	42.3%	14	11.9%
Diversity	34	26.3%	21	23.1%	24	20.3%
Gender	76	64.4%	27	38.5%	90	76.3%
Identity	41	28.8%	6	8.65%	13	11.0%

2.2 Automated Categorization

In the next phase of the study, the focus was on automation. In order to automate the search of relevant documents, as well as find a way of roughly quantifying the prevalence of a topic in a document before it is read, the following algorithm was devised. It was written in the Python programming language, and takes advantage of previous research into natural language processing and the modeling of word meaning undertaken at Stanford University. A set of target words are used as input, alongside a list of documents for which the researcher wants a score. The ranking algorithm makes use of the GloVe vector model developed by Manning, Socher, and Pennington (Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation). GloVe vectors are 300 dimensional vectors that mathematically encode the meaning of words. We will use the following notation to describe vectors corresponding to words in the encoded vocabulary:

$$i_{th}\text{word in vocabulary} = v_i \in \mathbb{R}^{300}$$

One of the most important properties of GloVe vectors for our algorithm is the proximity of synonyms in the vectors space. Generally, for any set of vectors v_i, v_j, v_k in the vocabulary, if the the inequality of euclidean distances

$$||v_i - v_j||_2 < ||v_i - v_k||_2$$

holds, then the j_{th} word in the vocabulary is more similar in meaning to the i_{th} word than the k_{th} word. The algorithm makes use of this property to rank documents based on the euclidean distances of the words it contains.

Keyword	JEE (total papers 2006-2016)	Percent with Keyword in JEE	IJEE (total papers 2006-2016)	Percent with Keyword in IJEE	JWMSE (total papers 2006-2016)	Percent with Keyword in JWMSE
Culture	52	34.7%	39	42.3%	14	11.9%
Diversity	34	26.3%	21	23.1%	24	20.3%
Gender	76	64.4%	27	38.5%	90	76.3%
Identity	41	28.8%	6	8.65%	13	11.0%

2.2 Automated Categorization

In the next phase of the study, the focus was on automation. In order to automate the search of relevant documents, as well as find a way of roughly quantifying the prevalence of a topic in a document before it is read, the following algorithm was devised. It was written in the Python programming language, and takes advantage of previous research into natural language processing and the modeling of word meaning undertaken at Stanford University. A set of target words are used as input, alongside a list of documents for which the researcher wants a score. The ranking algorithm makes use of the GloVe vector model developed by Manning, Socher, and Pennington (Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation). GloVe vectors are 300 dimensional vectors that mathematically encode the meaning of words. We will use the following notation to describe vectors corresponding to words in the encoded vocabulary:

$$i_{th}\text{word in vocabulary} = v_i \in \mathbb{R}^{300}$$

One of the most important properties of GloVe vectors for our algorithm is the proximity of synonyms in the vectors space. Generally, for any set of vectors v_i, v_j, v_k in the vocabulary, if the the inequality of euclidean distances

$$||v_i - v_j||_2 < ||v_i - v_k||_2$$

holds, then the j_{th} word in the vocabulary is more similar in meaning to the i_{th} word than the k_{th} word. The algorithm makes use of this property to rank documents based on the euclidean distances of the words it contains.

Now we will define how the similarity score for a document is calculated. For any document, the frequency of all words present in the document is recorded. Next, we define the distance of any one word to the set of target words as follows:

$$dist(v_i, targets) = \quad argmin_{t \in targets}\{||v_i - t||_2\}$$

where v_i is the GloVe vector of the current word and t is the vector corresponding to the target word. We calculate and record this distance for every unique word in the document. Next, we look at all the words for whom we have calculated this distance score and extract the top 15 "closest" unique words in the document, where closeness is measured by the distance metric. The number 15 was chosen empirically from testing the algorithm; a higher number would in some cases cause the algorithm to give too much weight to "filler" words (such as "and", "or", "such") or other common words that many documents might share.

Lastly, we assign the document a "score" with the function:

$$\frac{1}{\sum_{i=1}^{n} count(v_i)} \sum_{i=1}^{n} count(v_i) * dist(v_i, targets)$$

where n is 15, and v_1, v_2,...,v_{15} are the word vectors corresponding to the closest 15 unique words. count(v_i) is simply the number of occurrences of the word corresponding to vector v_i in the document, so that words that occur more frequently are given more weight when calculating score.

The rationale behind this scoring metric is that documents containing many synonyms or words closely related to the target words are more likely to have words whose vectors are close in euclidean distance to the vectors of the target words, and will thus have a lower sum of these distances in the equation above. Documents are then ranked based on the scoring metric, where lower scores are considered more closely related to the target vectors.

It is important to note that because the GloVe vectors were developed using Common Crawl, a dataset of text taken from the internet, some biases that may be present in the data set may also carry over into the model. For example, words like "engineer", "scientist", and "programmer" may be considered by the model to be closer in meaning to "man" than they are to "woman". This is due to the model picking up on gender disparities present in the frequency in which the words are used to describe men as opposed to women. There is currently some work being done to debias word embeddings (https://arxiv.org/abs/1607.06520), and future improvements to the algorithm will likely incorporate such debiasing.

The final score generally ranges between 0 and 9, with lower scores indicating that a paper is more likely to be related to the target word. While the scores are effective as comparative measures of relevance between papers, they can also be used to classify a document as relevant or not. Empirically, we found that scores below five tend to indicate that a paper is very likely to be relevant to a given topic, and would be classified as 'quite relevant'. A score between 5 and 6 was classified as 'slightly relevant', and generally indicated a paper that either mentioned the topic briefly or was generally about a topic tangential or slightly related to the target topic or word. A score close to 7 was found to indicate a 'neutral' paper, one which does not cover the topic or mentions it and topics related to it very sparsely. Scores between 6 and 7 were considered borderline, as they could indicate very brief mentions of the topic in a paper, but borderline papers were also likely to have very little relevance to the target topic.

To illustrate the difference between quite relevant classifications and slightly relevant classifications, as well as show the algorithm's results, we will use the example of running the algorithm on a group of documents with the target topic of culture. The algorithm performs a weighted average of the distance of the 15 words in the document with the smallest vector distance from the target word, so documents containing the word culture, or words close to culture in meaning, will rank more highly than documents that mention the topic with less frequency, or use words further in meaning. Below is the percentage of documents containing the word "culture", as well as the percentage of documents containing other words of varying closeness in meaning to culture, across both the quite relevant' and slightly relevant classifications.

Table 2: Percentage of keywords used when flagging a paper for "culture"

Word	**Percentage quite relevant mentions**	**Percentage slightly relevant mentions**
culture	100	98.08
cultural	97.22	80.77
society	88.89	59.62
social	86.11	90.38
cultures	83.33	67.31
literature	75	75
importance	72.22	82.69
sense	72.22	69.23
Culture	66.67	44.23

perspective	66.67	78.85
diversity	63.89	63.46
influence	58.33	69.23
attitudes	50	48.08
context	47.22	65.38
history	41.67	44.23
influenced	36.11	44.23
especially	33.33	28.85
multicultural	25	21.15
perspectives	25	36.58
traditions	25	19.23
evident	22.22	13.46

Quite relevant documents are very likely to contain the word “culture” or “Culture” (the algorithm is case sensitive, hence the distinction), as well words with similar meanings, such as “culture”, “cultures”, “society”, “social” and “diversity”. They also occasionally contain words such as “history” or “perspective” - words that have some overlap in meaning but are not direct synonyms. Slightly relevant documents are still likely to contain the word “culture”, but are also more likely to contain words such as “perspective”, “context”, and “importance”: words that may appear in similar contexts to “culture”, but have less similarity in meaning. Because slightly relevant documents are less likely to contain direct synonyms, and more likely to contain more distant words, they less likely to talk about culture directly; however they may still discuss topics that are somewhat closely related to culture, such as “history”, which appears in slightly relevant papers more than it does in quite relevant papers.

3 Results

3.1 Case 1: Assessing literature with automated ranking

The first case compares the automated ranking to the original manual categorization of journal papers published in JEE performed by Chloe. The overall distribution of ranks for each paper is shown in the figure below where n is the sample size..

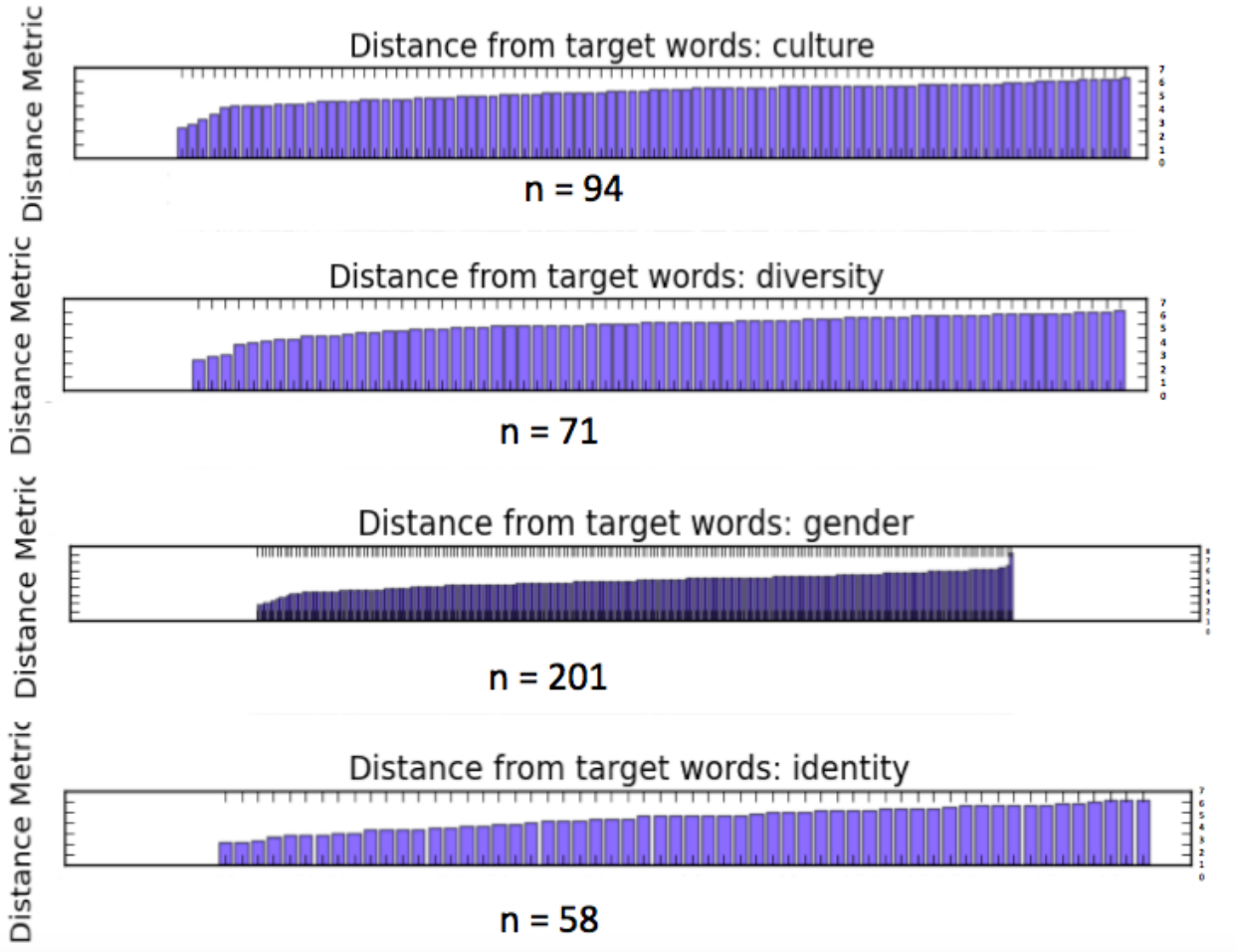

Figure 1: Distributions of papers for each target word

There is some difference between what manual coding and the Document Ranking Algorithm found to be relevant. The Document Ranking Algorithm found about half of the papers we found substantive to be 'quite relevant.' There was significantly more agreement between the methods when the Document Ranking Algorithm included papers that were 'slightly relevant.' The agreement between hand coding and quite relevan' papers ranged from 36.5% to 62.4% with culture having the lowest agreement and identity having the highest agreement. The agreement between manual coding and papers that were either quite relelvant or slightly relevant ranged from 83.6% to 98.2% with diversity having the lowest agreement and identity having the highest agreement. The breakdown by keyword is shown in the table below. Generally, the Document Ranking Algorithm classified the papers picked by manual coding that were neither quite relevant nor slightly relevant as 'borderline'. There was one paper classified as probably not relevant, and it was in the batch of gender papers.

Table 3: Comparison of Manual Coding with Document Ranking Algorithm Categorization

Keyword	Number of papers identified as 'quite relevant'	Percentage Difference between DRA and Manual Coding	Number of papers identified as 'quite relevant' or 'slightly relevant'	Percentage Difference between DRA and Manual Coding
Culture	37	-60.64	89	-5.32
Diversity	32	-54.29	69	-1.43
Gender	113	-44.06	190	-5.94
Identity	35	-39.66	54	-6.90

During manual coding, we flagged many papers for multiple keywords. This was corroborated by the Data Ranking Algorithm, which used similar words to calculate scores between keywords ("diversity" being used as a word to calculate the score for papers on culture, "identity" being used to calculate the score for papers on gender, etc.). Automation relied more heavily on word counts and was more thoroughly able to use context by using related words to make a calculation. The use of filler words to calculate a score was somewhat present ("importance" was used for about 70% of quite relevant papers on culture and "especially" was used for about 30%) but didn't appear to hamper the results.

3.2 Case 2: Assessing document data with automated ranking

The second case is based on another student (Benedikt von Unold) in the 2nd offering of EEL seeing the potential of the algorithm in researching how culture and gender are included in engineering product development. He used a qualitative research approach. Here, and especially in case study research, it is often hard to find out which texts fit best to answer the research question or to best explore the research topic. Researchers are often overwhelmed by a mass of qualitative data. In this case, Benedikt faced a document data collection of more than 17.000 pages of design reports. These had to be studied to first find the right cases for exploring peopleness considerations in engineering design projects and to then analyze the identified cases.

Particularly for the first task, the identification of cases, the program supports a faster, more accurate and objective process. The latter is especially helpful since in case studies the selection of cases is an important aspect but often based on subjective decisions. Research (Eisenhardt, 1989; Jin 2003) proposes several strategies such as using polar cases or extreme cases, nevertheless most cases are too complex and multifaceted to be ranked or even compared. In fact, case study research is often applied to explain causal links in phenomenon that are too complex for other research methods like surveys. The Document Ranking Algorithm can contribute to solve this dilemma. It is capable of sorting texts into rich and poor sources by

showing how much of a topic is addressed in a given text.
In the case of Benedikt's research, key insights could be deduced in an early stage, such as that his cases do not include a lot of information regarding gender, but that there are cases containing a lot of data about the topic elderly (see Figure 2). Hence, extreme cases could be identified to analyze how design teams consider the needs of elderly whereas these cases are inappropriate to analyze the topic gender in design.

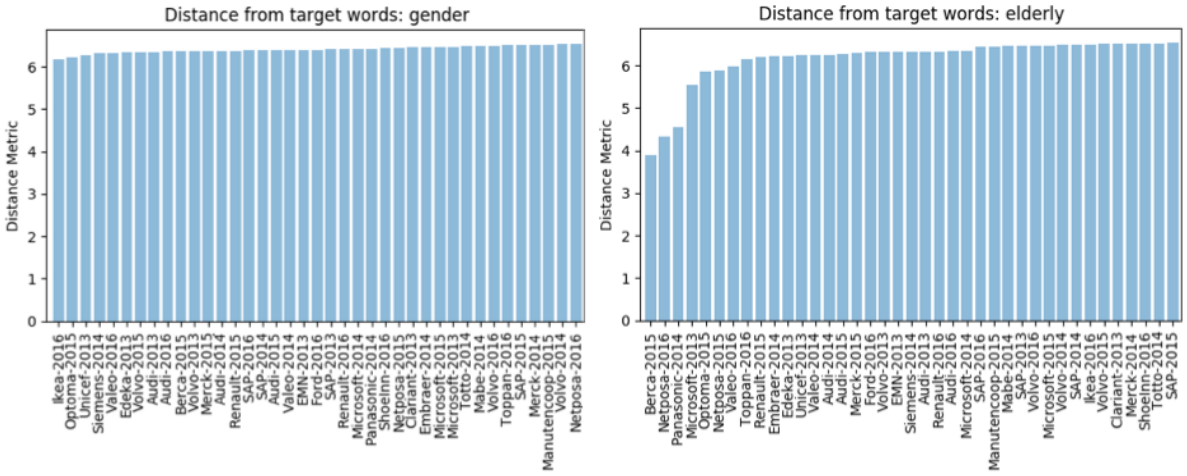

Figure 2. Ranked distance of the document data to the concepts of "gender" and "elderly"

Furthermore, a list of background characteristics (such as culture, gender, socio-economic class, etc.) was created and an average vector for all of these characteristics was calculated for every project. In this way, it was possible to see which projects were successful in considering a broad range of characteristics and which rather not. This information was used to strengthen the case selection that was done manually with a team of design experts before using the algorithm.

In both application examples, the Data Ranking Algorithm provided information that allowed distinct decisions that were grounded in real data. This helps researchers to steer in an early stage of a case study where there is normally no clear right or wrong. In this research, the Data Ranking Algorithm was mostly used to confirm critical decisions.

3.3 Case 3: Assessing interview data with automated ranking

In the final case, the Data Ranking Algorithm was applied to interview transcripts in an exploration of the use of algorithm-generated association strengths and concepts in comparison to interviewee self-assessment and traditional qualitative thematic coding. At its best, automated ranking could potentially enrich qualitative coding through suggesting subtle underlying connections to concepts, as well as enable combing through larger amounts of data.

In this case, the assessed data consisted of 35 interview transcripts (totalling in 367 pages) of early career engineers describing their experiences at their workplace. (These had been produced for a qualitative substudy led by Dr. Björklund within the Engineering Majors Survey research project.) In the beginning of the interview, the interviewees were asked whether they considered their current position as 1) innovative and 2) engineering. These reported self-assessments were then compared against the ranking produced by the developed algorithm according to the target words of innovation and engineering. The generated ranking bore little resemblance to the self-assessments of the interviewees (see Fig. 3, below, for a comparison with the concept of innovation).

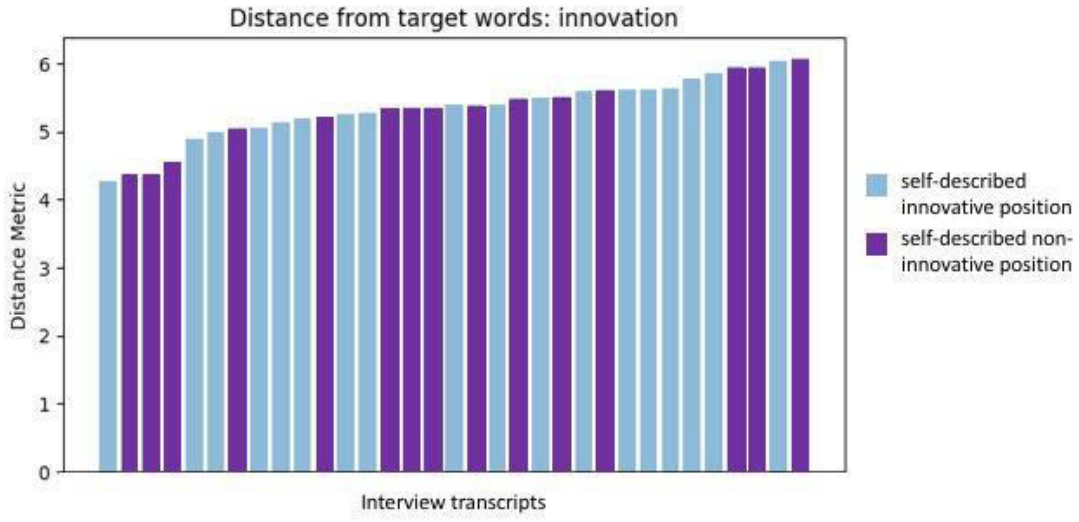

Figure 3. Ranked distance in comparison to engineers' self-assessment of "innovation"

Next, the five most closely related concepts reported per transcript with the target word of "innovation" and "engineering" were examined. The five most frequently reported concepts for innovation within the transcripts were "innovative", "technology", "industry", "initiative" and "future", and the most frequently reported highest related concepts for engineering were "engineer", "technical", "designing", "mechanical", "knowledge" and "project".

The top five related concepts were compared to the strength of the ranked association to innovation and engineering, with no pattern detected. However, comparing most related concepts between self-described innovative and non-innovative positions demonstrated more connection to qualitative assessment: While both groups shared "technology", "development", "industry" and "focus" as four out of the five most closely related concepts for innovation within the transcripts, but those in self-described innovative position had "success" as the fifth most related concept, whereas those in non-innovative positions had "initiative". Indeed, limited opportunities and negative experiences were more commonly reported by those in self-described non-

innovative positions, and those self-described innovative positions were more likely to report satisfaction and plans for continuing on their current career path.

Clearly, automated assessment cannot substitute human qualitative judgement in its current form. However, due to the ease and speed of ranking compared to the lengthy and labor-intensive process of manual qualitative content coding, related concepts can be explored to multiple target concepts as a point of inspiration and reflection within the qualitative analysis process.

4 Conclusion

Our paper concludes by summarizing the limits of automation and the potential of automation. The Document Ranking Algorithm currently flags presence of key words, but that may not correlate to content. Therefore, keywords need to be carefully selected so that it finds the papers most useful to the researcher. The automation process is sensitive to the keywords and to what levels are selected as to text being relevant to the keyword. The Document Ranking Algorithm is not yet a substitute for handcoding but can play a complementary role in helping flag text that is rich in including various factors (so as to help a researcher focus his/her attention) and/or in confirming manually coded text.

Appendix

Manual Coding Methodology

We used each publication's database to search for papers using the keywords culture diversity, gender, and identity. We looked through the list of articles that the database search engine pulled up. Most search engines showed the searched word highlighted in the text. In those instances, we were able to get a rough count of how frequently the word was used and where in the paper the word was used. The general criteria for flagging a paper was that the paper used the keyword at least three times. Papers that only used the keyword in the introduction or conclusion were excluded because these papers often only used the keyword as an application (for example, speculating that this research could be used to increase diversity of engineering students or seeing problems with engineering culture that motivated them to research their main topic). Some databases didn't highlight the word, so those papers were downloaded and searched in PDF form or their keywords, methodology, and results sections were skimmed.

Manual for Using the Data Ranking Algorithm

We then converted the papers collected by manual coding into txt files, ran the Document Ranking Algorithm on the files, and analyzed the distribution of their overall scores. Ranks are from 0 – 9 with lower scores showing that the word is more relevant. Papers with a score of 5 or

under were ‘quite relevant’, between 5 and 6 were ‘slightly less relevant,’ 6 – 7 were ‘borderline’, and 7 and up were ‘probably not relevant’. We looked at the percentage of papers that the Document Ranking Algorithm found to have varying degrees of relevance. We also noted the related words that the Document Ranking Algorithm used to calculate the rank for each paper and looked at the difference in words for papers that the Document Ranking Algorithm found to be ‘quite relevant’ and papers getRank.py found to be ‘slightly relevant.’ It was suggested that the Document Ranking Algorithm be run on every paper published in the target publications from 2006 - 2016 to see if the Document Ranking Algorithm found relevant papers that hand coding did not, but this idea was not pursued due to time constraints.

To run the algorithm, create a file for all things related to the program. Within this file, there should be a file for the txt files, a file for the files that the Document Ranking Algorithm will create detailing the scores of the papers and the words used to calculate the score, and a file for the distribution graph of scores that the Document Ranking Algorithm will create. The code is modified to reflect the keywords being investigated and the file names then run on the computer’s terminal. For the batches in this research, it took about five minutes for the Document Ranking Algorithm to produce the report and distribution graph.

References

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of management review, 14*(4), 532-550.

Nikhil Garg, Londa Schiebinger, James Zou, Word Embeddings Quantify 100 Years of Gender and Ethnic Stereotypes, manuscript under review/revision (Nov. 2017)

Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, Adam Kalai, Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings, 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

Levine, A. (2016). XXXX. Unpublished manuscript.

Pawley, A. L., Schimpf, C., & Nelson, L. (2016). Gender in engineering education research: A content analysis of research in JEE, 1998–2012. Journal of Engineering Education, 105(3).

Yin, R. K. (2003). *Case study research: Design and methods*. Thousands Oaks: Sage publications.

Now we will define how the similarity score for a document is calculated. For any document, the frequency of all words present in the document is recorded. Next, we define the distance of any one word to the set of target words as follows:

$$dist(v_i, targets) = \ argmin_{t \in targets}\{||v_i - t||_2\}$$

where v_i is the GloVe vector of the current word and t is the vector corresponding to the target word. We calculate and record this distance for every unique word in the document. Next, we look at all the words for whom we have calculated this distance score and extract the top 15 "closest" unique words in the document, where closeness is measured by the distance metric. The number 15 was chosen empirically from testing the algorithm; a higher number would in some cases cause the algorithm to give too much weight to "filler" words (such as "and", "or", "such") or other common words that many documents might share.

Lastly, we assign the document a "score" with the function:

$$\frac{1}{\sum_{i=1}^{n} count(v_i)} \sum_{i=1}^{n} count(v_i) * dist(v_i, targets)$$

where n is 15, and v_1, v_2,...,v_{15} are the word vectors corresponding to the closest 15 unique words. count(v_i) is simply the number of occurrences of the word corresponding to vector v_i in the document, so that words that occur more frequently are given more weight when calculating score.

The rationale behind this scoring metric is that documents containing many synonyms or words closely related to the target words are more likely to have words whose vectors are close in euclidean distance to the vectors of the target words, and will thus have a lower sum of these distances in the equation above. Documents are then ranked based on the scoring metric, where lower scores are considered more closely related to the target vectors.

It is important to note that because the GloVe vectors were developed using Common Crawl, a dataset of text taken from the internet, some biases that may be present in the data set may also carry over into the model. For example, words like "engineer", "scientist", and "programmer" may be considered by the model to be closer in meaning to "man" than they are to "woman". This is due to the model picking up on gender disparities present in the frequency in which the words are used to describe men as opposed to women. There is currently some work being done to debias word embeddings (https://arxiv.org/abs/1607.06520), and future improvements to the algorithm will likely incorporate such debiasing.

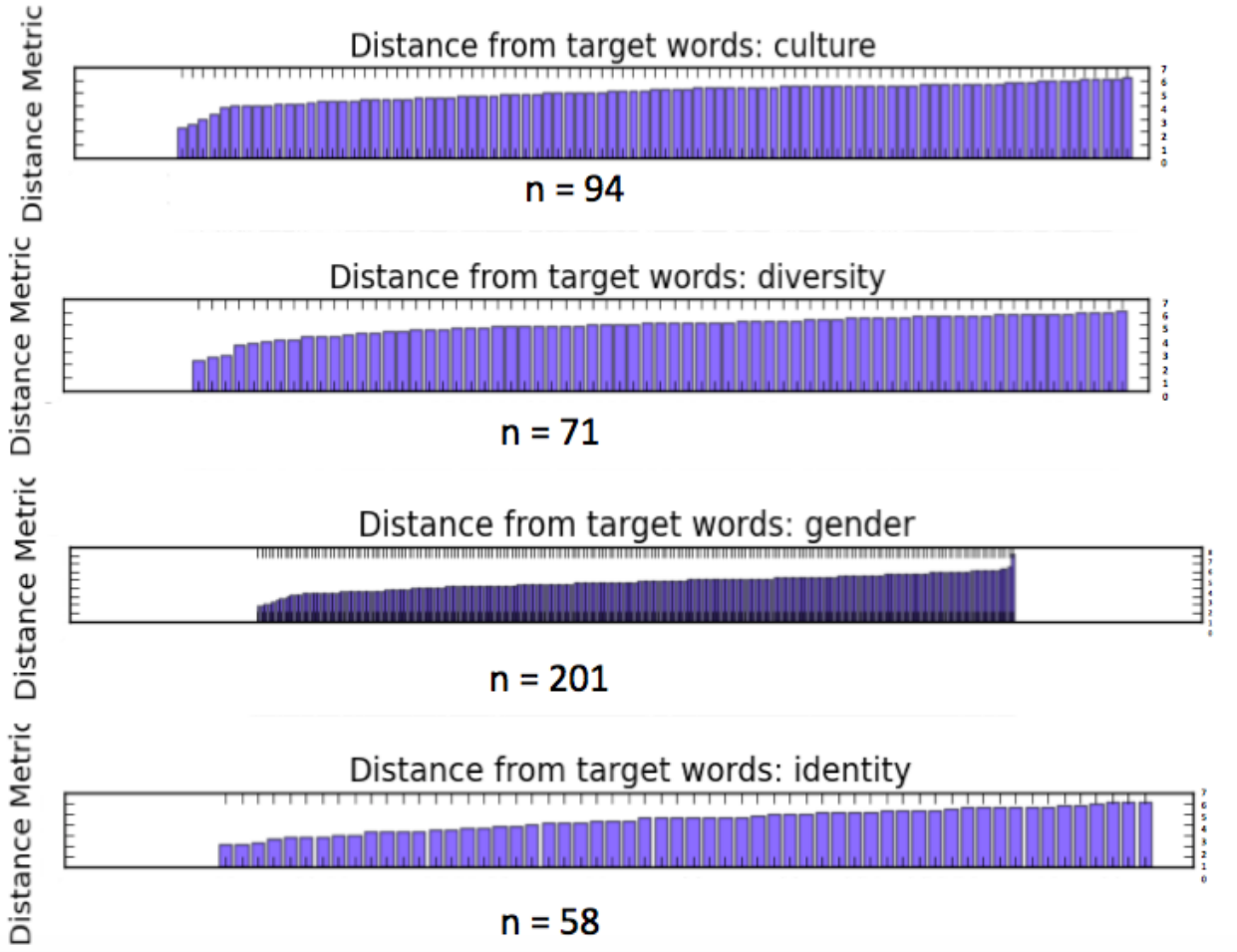

Figure 1: Distributions of papers for each target word

There is some difference between what manual coding and the Document Ranking Algorithm found to be relevant. The Document Ranking Algorithm found about half of the papers we found substantive to be 'quite relevant.' There was significantly more agreement between the methods when the Document Ranking Algorithm included papers that were 'slightly relevant.' The agreement between hand coding and quite relevan' papers ranged from 36.5% to 62.4% with culture having the lowest agreement and identity having the highest agreement. The agreement between manual coding and papers that were either quite relelvant or slightly relevant ranged from 83.6% to 98.2% with diversity having the lowest agreement and identity having the highest agreement. The breakdown by keyword is shown in the table below. Generally, the Document Ranking Algorithm classified the papers picked by manual coding that were neither quite relevant nor slightly relevant as 'borderline'. There was one paper classified as probably not relevant, and it was in the batch of gender papers.

Table 3: Comparison of Manual Coding with Document Ranking Algorithm Categorization

showing how much of a topic is addressed in a given text.
In the case of Benedikt's research, key insights could be deduced in an early stage, such as that his cases do not include a lot of information regarding gender, but that there are cases containing a lot of data about the topic elderly (see Figure 2). Hence, extreme cases could be identified to analyze how design teams consider the needs of elderly whereas these cases are inappropriate to analyze the topic gender in design.

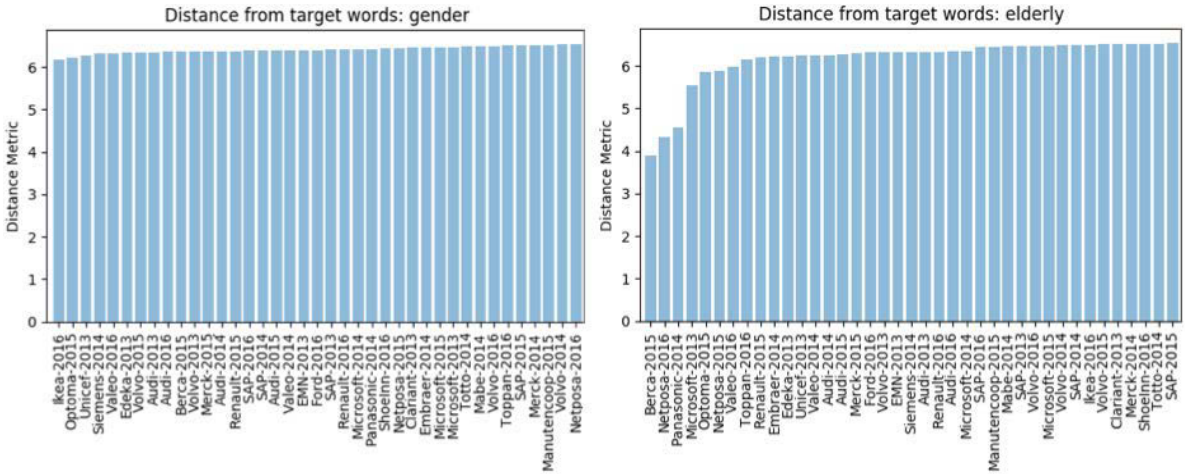

Figure 2. Ranked distance of the document data to the concepts of "gender" and "elderly"

Furthermore, a list of background characteristics (such as culture, gender, socio-economic class, etc.) was created and an average vector for all of these characteristics was calculated for every project. In this way, it was possible to see which projects were successful in considering a broad range of characteristics and which rather not. This information was used to strengthen the case selection that was done manually with a team of design experts before using the algorithm.

In both application examples, the Data Ranking Algorithm provided information that allowed distinct decisions that were grounded in real data. This helps researchers to steer in an early stage of a case study where there is normally no clear right or wrong. In this research, the Data Ranking Algorithm was mostly used to confirm critical decisions.

3.3 Case 3: Assessing interview data with automated ranking

In the final case, the Data Ranking Algorithm was applied to interview transcripts in an exploration of the use of algorithm-generated association strengths and concepts in comparison to interviewee self-assessment and traditional qualitative thematic coding. At its best, automated ranking could potentially enrich qualitative coding through suggesting subtle underlying connections to concepts, as well as enable combing through larger amounts of data.