STATISTICS · DATA SCIENCE AND MACHINE LEARNING

Text Analytics and Natural Language Processing for Open-Ended Responses

How text analytics turns open-ended survey answers into reliable numbers, from word counts and topic models to supervised classifiers and language models, and how to validate the results.

ModQu research article · 1,297 words · 13 references (APA 7th edition)

Open-ended questions let respondents answer in their own words. They capture reasons, frustrations and ideas that a fixed list of options would miss, and they guard against the researcher's assumptions shaping every answer (Krosnick, 1999). The price is analysis: a study of 5,000 customers can produce 5,000 short paragraphs. **Text analytics**, also called natural language processing (NLP) when the emphasis is on computational methods, provides ways to turn that text into structured data that can be counted, compared across groups and tracked over time.

Text as data

Every automated method begins by representing text as numbers. Grimmer and Stewart (2013) set out four principles that remain the best guide for applied work: all quantitative models of language are wrong, but some are useful; automated methods amplify human reading rather than replace it; there is no single best method for all tasks; and every method must be validated against human judgement. Gentzkow et al. (2019) reach similar conclusions for business and economic text.

The simplest representation is the **bag of words**. Each response becomes a row in a document-term matrix, with one column per word and cells holding counts. Word order is discarded, which sounds crude but works surprisingly well for sorting short answers into themes.

Preprocessing

Before counting, analysts usually lower-case the text, remove punctuation, drop very common "stop words" such as *the* and *and*, and reduce words to a common stem so that *cancel*, *cancelled* and *cancelling* are counted together (Manning et al., 2008). These choices are not neutral. Denny and Spirling (2018) showed that different reasonable preprocessing steps can change the results of unsupervised analyses substantially, so the steps should be documented and, where results matter, tested for sensitivity.

Weighting words

Raw counts give too much weight to words that appear everywhere. The **tf-idf** weight, which builds on Sparck Jones's (1972) idea of term specificity, multiplies a word's frequency in a response by how rare it is across all responses:

$$\text{tf-idf}(t, d) = \text{tf}(t, d) \times \log(N \div \text{df}_t)$$

Here tf(t, d) is the count of term *t* in response *d*, *N* is the number of responses and df_t is the number of responses containing *t*. A word such as *price* that appears in a third of answers receives a modest weight; a word such as *overcharged* that appears in a few receives a higher one. Salton and Buckley (1988) compared many such weighting schemes and found that this combination of local frequency and global rarity performs well.

Four families of methods

Dictionaries and keyword rules. The analyst lists words linked to each theme and counts matches. This is transparent and cheap, but it misses synonyms, misspellings and context, and dictionaries built for one setting often transfer poorly to another (Grimmer & Stewart, 2013).

Supervised classification. Human coders label a random sample of responses using a codeframe, and a statistical model such as regularised logistic regression learns to predict those labels from the text. The model then codes the remaining responses. This is the closest automated equivalent to traditional coding, and its accuracy can be measured directly on held-out labelled responses.

Unsupervised topic models. Methods such as latent Dirichlet allocation treat each response as a mixture of topics and each topic as a distribution over words, estimated without any labels (Blei et al., 2003). The structural topic model extends this by letting topic prevalence depend on respondent characteristics such as age or region. Roberts et al. (2014) developed it specifically for open-ended survey responses. Topic models are useful for discovering themes no one anticipated, but the topics need careful human interpretation and naming.

Embeddings and language models. Word embeddings represent each word as a dense vector learned from large text collections, so that words used in similar contexts lie close together (Mikolov et al., 2013). Pre-trained transformer models such as BERT go further and represent each word in the context of its sentence (Devlin et al., 2019). Fine-tuned on a few hundred coded responses, such models often classify short, informal text more accurately than bag-of-words methods, because they recognise that "too dear" and "not worth the money" express the same idea. They are harder to explain, and their errors are less predictable.

Large generative language models can also be prompted to assign codes without any training examples. This can speed up a first draft of the codeframe, but the same validation applies: their codes should be compared with human coding on a random sample, and the prompt, model version and settings should be recorded so the coding can be reproduced.

A worked example

Consider a hypothetical subscription service that asks 2,400 former customers, "What is the main reason you cancelled?" Two analysts read a random sample of 400 answers and develop a codeframe of six themes, in the spirit of thematic analysis (Braun & Clarke, 2006). Both code 150 of the same responses independently, and their agreement gives Cohen's kappa of 0.85, which Landis and Koch (1977) would class as almost perfect agreement.

A supervised classifier is then trained on 300 coded responses and tested on the other 100. It matches the human code for 86% of the test responses, with a kappa of 0.81. Applied to all 2,400 answers, it produces theme shares close to those in the hand-coded sample, as Figure 1 shows. The team can now break the themes down by plan type, tenure and region, which would have been impractical with hand coding alone.

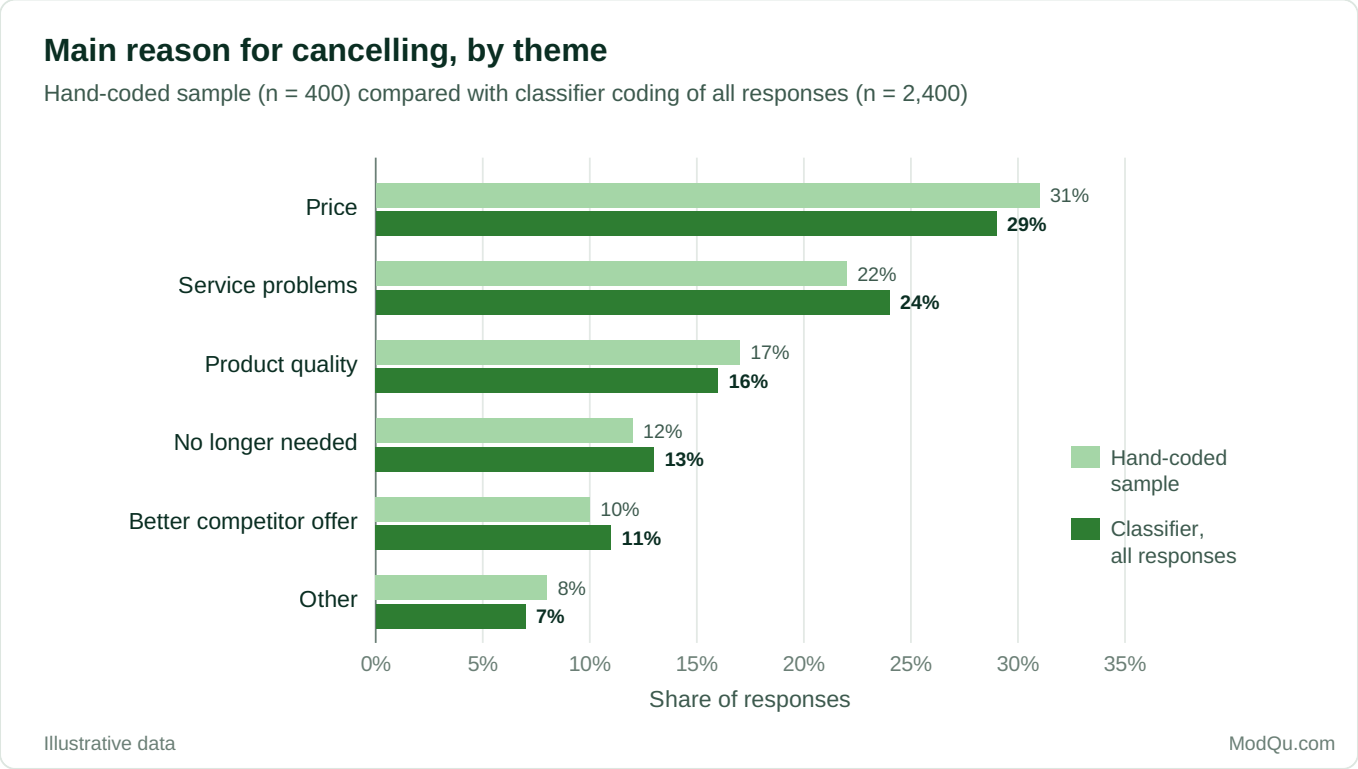


Figure 1. Share of cancellation reasons by theme, hand-coded sample compared with automated coding of all responses. Close agreement at the aggregate level supports, but does not prove, accuracy for individual responses. Illustrative data; chart by ModQu.

Practical guidance

- **Start with a codeframe and a human-coded sample.** Even unsupervised methods need a labelled sample for validation. Draw it at random, not from the first or most vivid responses.
- **Decide how codes combine.** If the question asks for one main reason, themes should be mutually exclusive; if respondents give several reasons, allow multiple codes per response. Decide this before coding starts, because it changes how shares add up.
- **Measure reliability twice.** Report agreement between human coders, then agreement between the model and humans on held-out data. A model cannot be more trustworthy than the codes it learned from.
- **Check accuracy by subgroup.** A classifier can work well overall and poorly for one region, language or age group whose wording differs.
- **Keep the verbatims.** Quotations make themes concrete for decision-makers and allow others to check the coding.

- **Re-validate over time.** New products, events and slang change how people write. A model trained last year should be tested on fresh responses before its trends are reported.

Common mistakes

- **Treating topics as findings.** A topic model's output is a set of word lists. Naming a topic "price concerns" is an interpretation that must be checked by reading responses.
- **Ignoring response length and nonresponse.** Respondents who write long answers are not typical of all respondents, and many skip open questions entirely. Shares should be reported relative to a clear base.
- **Over-trusting a single accuracy figure.** Overall accuracy can hide poor performance on small but important themes. Report results for each theme.
- **Using the model's own training data to judge it.** Accuracy must be measured on responses the model has not seen.

Key takeaways

Text analytics makes open-ended responses usable at scale. Choose the simplest method that answers the question: dictionaries for narrow, well-defined concepts, supervised classifiers for a known codeframe, topic models for discovery and fine-tuned language models when wording is varied and informal. Whatever the method, its output is only as credible as its validation against careful human coding.

References

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology, 3*(2), 77–101. https://doi.org/10.1191/1478088706qp063oa

Denny, M. J., & Spirling, A. (2018). Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. *Political Analysis, 26*(2), 168–189. https://doi.org/10.1017/pan.2017.44

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Gentzkow, M., Kelly, B., & Taddy, M. (2019). Text as data. *Journal of Economic Literature, 57*(3), 535–574. https://doi.org/10.1257/jel.20181020

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis, 21*(3), 267–297. https://doi.org/10.1093/pan/mps028

Krosnick, J. A. (1999). Survey research. *Annual Review of Psychology, 50*(1), 537–567. https://doi.org/10.1146/annurev.psych.50.1.537

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159–174. https://doi.org/10.2307/2529310

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press. https://doi.org/10.1017/CBO9780511809071

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.1301.3781

Roberts, M. E., Stewart, B. M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S. K., Albertson, B., & Rand, D. G. (2014). Structural topic models for open-ended survey responses. *American Journal of Political Science, 58*(4), 1064–1082. https://doi.org/10.1111/ajps.12103

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513–523. https://doi.org/10.1016/0306-4573(88)90021-0

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation, 28*(1), 11–21. https://doi.org/10.1108/eb026526