Detecting Bot-Answerable Questions in Ubuntu Chat

David C. Uthus[1,2,3] David W. Aha[2]

[1]National Research Council Postdoctoral Fellow
[2]Naval Research Laboratory
[3]Now at Google Inc.

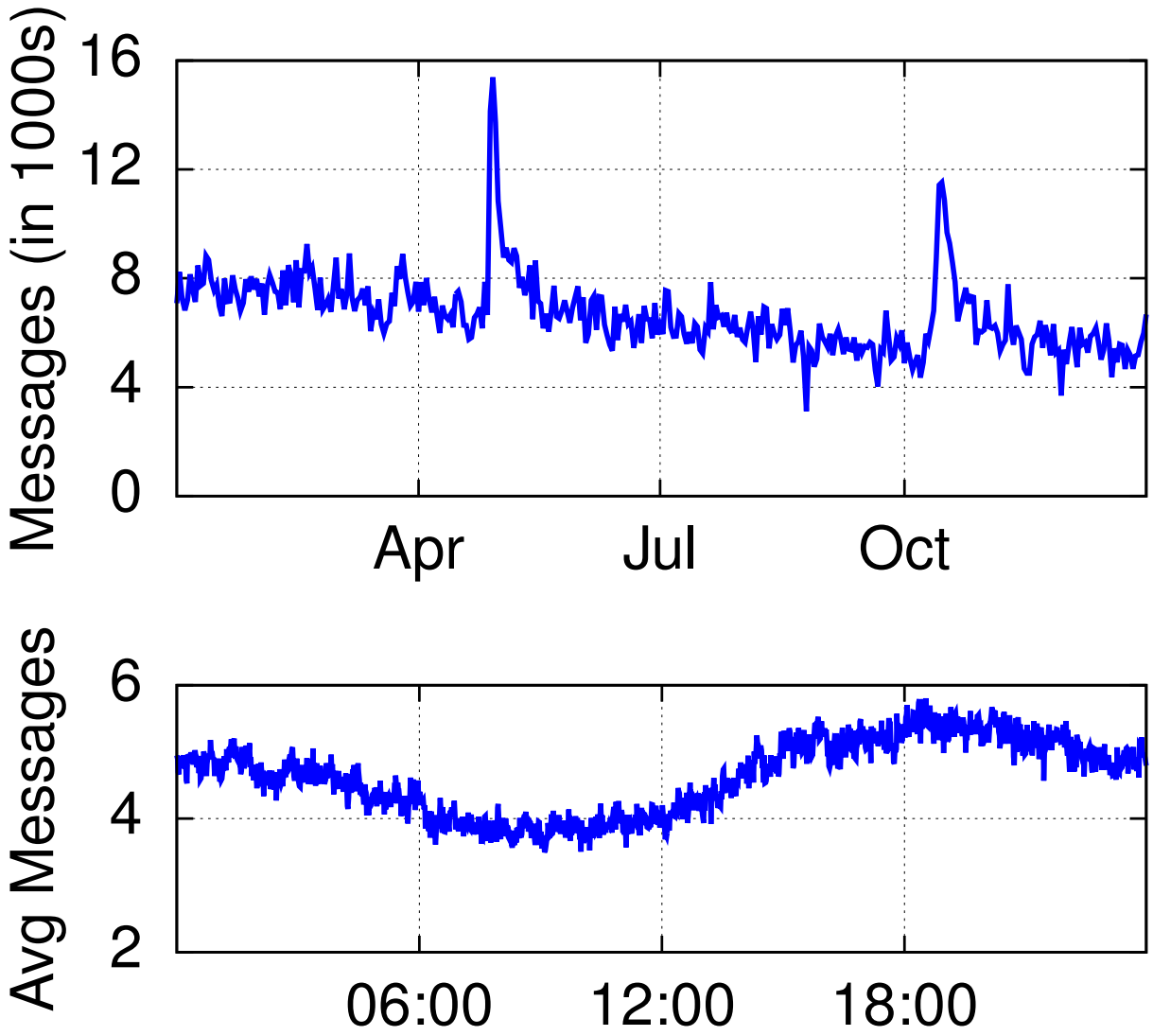
IJCNLP October 2013

Ubuntu's IRC Technical Support Channels

- Community-run, Ubuntu-supported real-time support
- Primary channel is `#ubuntu`
- Multiple topic- and language-focused channels available


```
[13:04] <adac> Does external software (software not
installed via package manager ), even web interfaces go to
/opt by default?
[13:04] <jrib> adac:  it goes where you want to put it.
Customary locations are /usr/local/ and /opt
```

#ubuntu's Traffic (2011)



ubottu – Ubuntu's IRC Channel Bot

- Found in most of Ubuntu's IRC support channels
- Contains a set of factoids (mapped to a set of factoid commands) for answering FAQs
- Must be **manually invoked** (often by experts)

```
[13:19] <p5yx> is the netbook remix not available anymore?
[13:20] <histo> !unr | p5yx
[13:20] <ubottu> p5yx:  Starting with Ubuntu 11.04, the
Ubuntu Netbook Edition is no longer being offered as a
separate install as Unity is now standard for all Ubuntu
desktop installs.
```

Other open-source communities also use similar bots

Detecting Bot-Answerable Questions

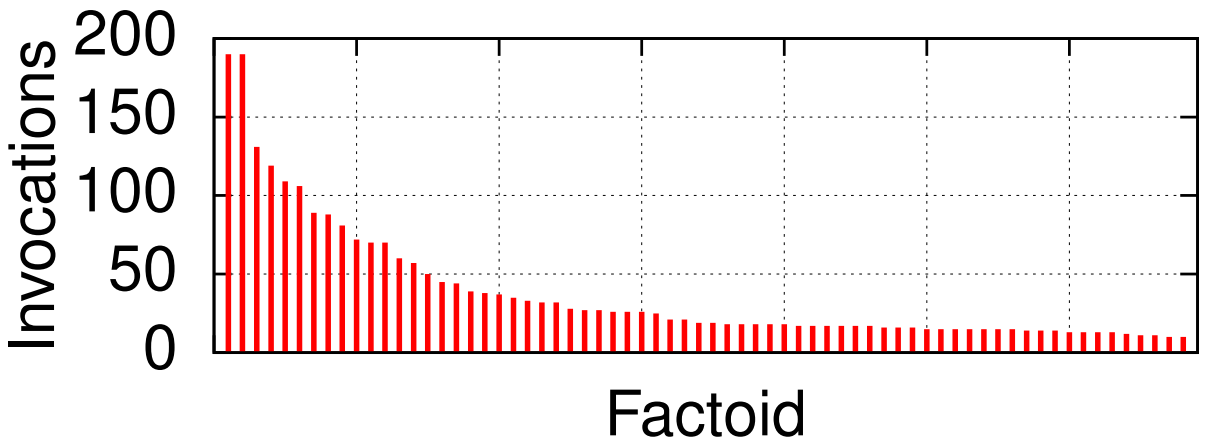
Long-term goal: Create an automated bot that can answer questions it is confident about and defer other questions to human experts.

Current goal: Can we automatically detect whether a question is bot-answerable (BAQ) (and which factoid to answer with) or human-answerable (HAQ) in a controlled environment?

Corpus

Manually labeled 4577 questions from the Ubuntu Chat Corpus (Uthus & Aha, 2013):

- 2002 HAQs – questions answered by bot experts
- 2575 BAQs – 68 factoid categories



- Available at `http://daviduthus.org/UCC/`

Approach – Baseline

Baseline – scan question for factoid command matches

- Similar to how humans do it
- Leads to wrong answers and angry users!

Approach – Learning Algorithms

Supervised learning on labeled data (Scikit-learn):

- *k*-NN
- SVM

Data representations:

- Bag-of-words
- Bigrams
- Character *n*-grams

tf $-$ *idf* to weigh features and χ^2 feature selection.

Metrics

10-fold cross validation evaluation protocol

Evaluation metrics:

- Precision
- Recall
- $F_{0.5}$ – emphasis on precision (boy who cried wolf. . .)

Results

SVM & *k*-NN outperformed baseline across all metrics
(Best $F_{0.5}$ scores – SVM 0.6, *k*-NN 0.49, baseline 0.37)

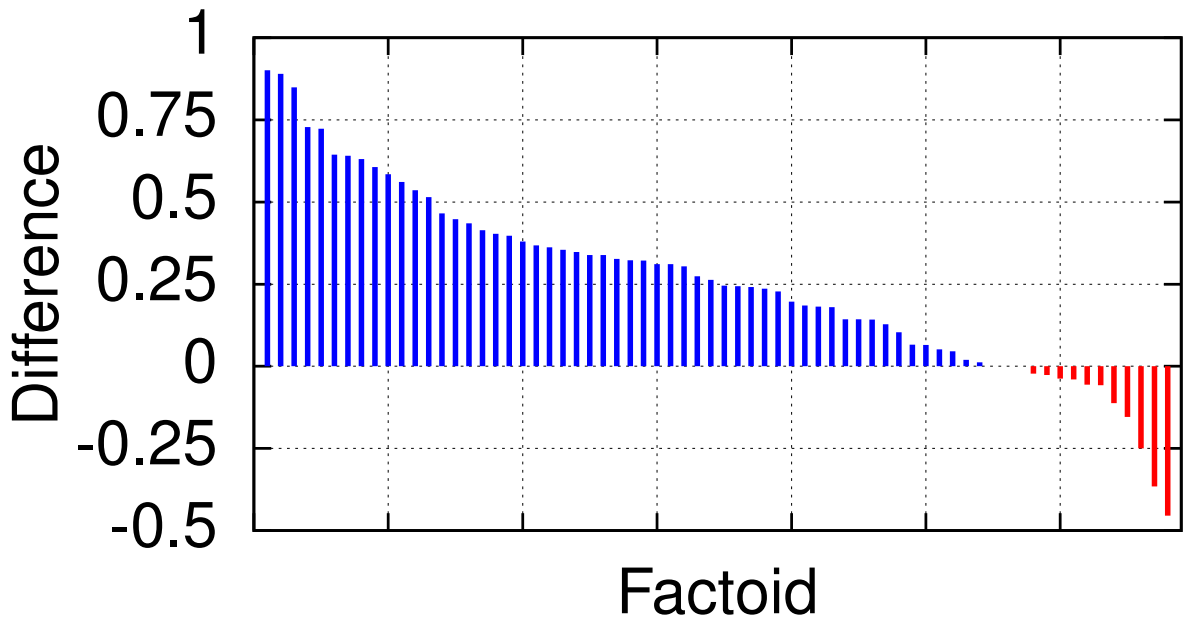


Figure: SVM vs baseline, comparison of $F_{0.5}$ scores per factoid.

Character *n*-grams offered better representation

Results – Questions

Learning algorithms did well on:

- Questions directing users to other channels

Learning algorithms struggled with:

- Questions which could be answered by similar factoids
  - `ask` vs `anyone`
- Questions covering a wide-range of topics
  - `#ubuntu`
  - `details`
  - `wine`

Conclusions

Contributions:

- Identify real-world problem
- Publicly-available corpus
- Initial empirical study on viability of applying learning algorithms
- Analysis of difficulty of question types

Future work:

- Apply unsupervised methods for finding more questions to match with the factoids
- Automatic generation of factoids through summarization

Thank you!

References I

Uthus, D. C., & Aha, D. W. (2013). The Ubuntu Chat Corpus for multiparticipant chat analysis. In *Proceedings of the AAAI Spring Symposium on Analyzing Microtext*. AAAI.