Pixie

Preference in Implicit and Explicit Comparisons

Amanul Haque
(Under the supervision of Dr. Munindar P. Singh)
Nov 12th, 2021

Department of Computer Science

NC STATE UNIVERSITY

Table of contents

1. Introduction and Motivation
2. Definitions and Problem Statement
3. New Dataset
4. Experiments and Results
5. Conclusion and future work

Intro

Comparison in Online Text

- Comparisons are prevalent in online text such as user reviews and blogs and are critical for mining arguments and business intelligence
    - Users often express their opinions (likes and dislikes) on a product by comparing it against its competitors
    - Consumers often rely on preferences of other consumers to make a purchasing decision (word-of-mouth)
    - Comparative reviews can be used to rank products and extract user expectations and comparative relations between products

Overlooked Comparative Constructions

- Prior works have focused on direct and explicit comparisons
  - e.g., *A* is better than *B*
- Comparative constructions overlooked by previous studies include
  - Omitted entity under comparison that can be inferred based on context (Omitted complements [8])
    - For example, The LED line of the printer A is clearly thicker *[than the one on B]*
  - Comparative constructions that lack explicit comparative linguistic cues (comparative quantifiers and superlatives)
    - For example, *A* is here immediately while *B* takes forever
- We found such comparisons common in user generated text like app reviews

What Are We Contributing?

- In this work we focus on
  - *Implicit comparisons* (one of the compared entity is omitted)
  - *Indirect comparisons* (comparisons that lacks comparative linguistic cues)
- We present Pixie, a manually annotated dataset for preference classification from app reviews
- We experiment with traditional and transformer-based ML models and compare our results with the SOTA in preference classification

Example Comparative Sentences from Pixie

Table 1: Examples of comparative sentences from reviews.

	Sentence	App
S_1	Bye ***Uber***, hello ***Lyft***.	*Uber*
S_2	Does ***this app*** really need to be 260 MB when the ***Marriott app*** is only 47 MB?	*Hilton Honors*
S_3	Beats the pants off ***pandora***.	*Spotify*

Definitions and Problem Statement

Comparison and Preference in Reviews

Identifying preference from reviews involves two tasks,

- Comparative Sentence Identification (CSI) [4]
  - Identifying comparative sentences from reviews
- Comparative Preference Classification (CPC) [3]
  - Identifying the preferred entity in a comparative sentence

Definition 2.1

A ***comparative sentence*** is defined as a sentence containing similarity, dissimilarity, or a preference between two entities.

Pixie includes

- *Explicit comparisons*: Both competing entities mentioned in the text (including pronominal references)
- *Implicit comparisons*: Only one of the competing entity mentioned in the text
- *Indirect comparisons*: Comparative sentences that lack explicit comparative linguistic structure or cues.

Table 2: Examples of types of Comparative Sentences

	Sentence	App	Comparison Type
S_1	If ***Uber*** had customer service that could be ***Lyft***	*Lyft*	Explicit, Indirect
S_2	I think that ***it's*** a lot more fun than ***temple Run***	*Subway Surfers*	Explicit, Direct
S_3	More info than ***cnbc*** app!	*Bloomberg*	Implicit, Direct

Definition 2.2

A ***preferred entity*** is defined as the entity chosen over the other based on an explicit or implicit preference revealed in a comparative sentence

A preferred entity can be,

- *Current* app (app being reviewed),
- *Other* app (competitor app), or
- *None* (i.e. ambiguous or no preference)

Table 3: Examples sentences showing preference.

	Sentence	App	Preferred Entity
S_1	Easy to use, more balanced than ***CNN*** and ***Wash Post***	*Fox news*	Current
S_2	I prefer the ***BBC app***.	*USA Today*	Other
S_4	Makes me want to switch back to ***Pandora***, but ***it's*** just as bad.	*Spotify*	None

Problem statement: Given a sentence,

$$s = (w_1, w_2, w_3, ..., w_n)$$

that contains a competing entity (other app) and may contain the current entity (app being review) mention, our goal is to identify the preferred entity between the two.

New Dataset

Dataset Creation

- Collected reviews for 179 popular apps on Apple App Store
- Manually grouped into 23 genres, including banking apps, airline apps, weather apps, communication apps, etc
- Apps within the same group are direct competitors (e.g., Instagram is competitor for Facebook and Snapchat)
- Tokenized app reviews into sentences and filtered sentences containing a competitor app mention
  - If a sentence mentions a competitor, it is likely to have a comparison
- Manually annotated the filtered sentences

- Extracted sentences are annotated for *comparison* and *preferred entity*.
    - Comparison:
        - *non-comparative, implicit comparison, explicit comparison*
    - Preferred entity:
        - *current app, other app, none*
- We then drop the non-comparative sentences and remove duplicate sentences.
- The final annotated version of Pixie contains 8,890 manually labeled comparative sentences.

Data Annotation (Inter-rater agreement)

Annotations for the dataset was conducted in three phases.

- Phase 1:
  - Authors (3 graduate students) labeled a sample dataset and discussed disagreements
  - Repeated three times to refine annotation instructions
- Phase 2:
  - 4,793 sentences annotated with each sentences labeled by two authors
  - Disagreements were resolved by the first author
  - Inter-rater agreement (Krippendorff alpha) was 0.82
- Phase 3:
  - 5,559 sentences labeled via crowdsourcing with 42 student participants
  - Each participant labelled 400 sentences and each sentence is labelled by three annotators
  - The final label chosen based on majority vote (first author breaks ties in cases when no clear majority)
  - Inter-rater agreement (Krippendorff alpha) was 0.74

Dataset Distribution

Table 4: Pixie Dataset Distribution

Preferred Entity	Comparison Type: Implicit	Comparison Type: Explicit	Total
Current	1910	2097	4007
Other	2199	1069	3268
None	758	857	1615
Total	4867	4023	8890

Masking

- Are there any problems with the annotated dataset?
  - Since we have limited app pairs in our dataset the model may learn to differentiate between classes based on app preference of users.
- Solution?
  - *Masking* is a mechanism to skip certain input tokens when processing the data by encoding those tokens with some predefined tag.
- We mask all entity mentions with two predefined tags,
  - current_app (for the apps being reviewed), and
  - other_app (for the competitor apps)
- Masking ensures that the model trained on Pixie learns the comparative and preference revealing linguistic semantics and not just the preference between the apps being compared.

Masked Sentences

Table 5: Masking app names in the sentence.

	Original sentence	Masked sentence
1	*CNN* should leave journalism to the pros at ***Fox*** news.	***<current_app>*** should leave journalism to the pros at ***<other_app>*** news.
2	way better than ***Pandora*** by a long shot!!!!	way better than ***<other_app>*** by a long shot!!!!
3	*This* is a great game just like ***Temple run***	***<current_app>*** is a great game just like ***<other_app>***

Experiments and Results

Experiments

- Prior Work (ED-GAT)
  - Entity-Aware Dependency-Based Deep Graph Attention Network (ED-GAT) [5]
  - Existing state-of-the-art for the task of CPC
- Traditional machine learning approaches
  - SVM, Random Forest, AdaBoost
- Transformer-based Approaches
  - Fine-tuning pretrained language models like BERT [2] and XLNET [10]
  - Our experiments include DistilBERT, RoBERTa, ALBERT, DeBERTa, and XLNET

- ED-GAT leverages a multi-hop Graph Attention Network (GATs) [9] to capture dependency relations in a sentence
- ED-GAT achieves a micro F1-score of 87.43% in identifying the preferred entity on the CompSent-19 dataset [6]
- We follow the format of the CompSent-19 dataset and convert all the sentences in Pixie dataset to match the formatting requirements.

Table 6: Results for ED-GAT on Pixie

Model	**CURRENT**			**NONE**			**OTHER**			**WEIGHTED AVERAGE**		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
ED-GAT	83.24	78.05	80.57	48.89	54.49	51.54	76.28	77.79	77.03	74.44	73.68	73.99

- We use SBERT (Sentence BERT) [7] for sentence embeddings
  - SBERT is pretrained BERT modified with Siamese and triplet network structures
  - SBERT achieves state-of-the-art results for five out of seven tasks on SentEval [1]

Table 7: Results for Traditional ML approaches on Pixie

Model	CURRENT			NONE			OTHER			WEIGHTED AVERAGE		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
AdaBoost	71.57	73.44	72.49	45.06	35.29	39.58	63.53	68.30	71.57	63.80	64.62	64.07
Random Forest	71.49	81.30	76.08	**64.80**	25.08	36.16	66.13	75.04	70.30	68.31	68.79	66.71
SVM	**76.99**	**82.17**	**79.49**	62.63	**36.84**	**46.39**	**71.04**	**79.63**	**75.09**	**72.19**	**73.00**	**71.86**

- We experiment with DistilBERT, RoBERTa, ALBERT, DeBERTa, and XLNET
- We adopt the AdamW Optimizer with a 5e-5 learning rate and a weight decay of 0.01 for fine-tuning.
- Each model is fine-tuned for 20 epochs on Pixie

Table 8: Results for Transformer-Based models

Model	CURRENT			NONE			OTHER			WEIGHTED AVERAGE		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
DistilBert_{base}	82.49	87.53	84.94	61.23	52.32	56.43	80.65	80.40	80.52	77.95	78.52	78.14
$\text{ALBERT}_{base\text{-}v2}$	87.30	87.41	87.35	58.86	57.59	58.22	82.70	83.46	83.08	80.44	80.54	80.49
$\text{XLNET}_{base\text{-}cased}$	85.80	91.15	88.39	66.03	53.56	59.15	83.73	85.15	84.43	81.45	82.11	81.63
RoBERTa_{base}	87.81	92.52	90.10	68.13	57.59	62.42	87.42	88.36	87.89	84.09	84.65	84.26
RoBERTa_{large}	**88.60**	**93.02**	**90.75**	**68.99**	**61.30**	**64.92**	**88.75**	88.21	**88.48**	**85.09**	**85.49**	**85.23**
DeBERTa_{base}	87.97	91.15	89.53	67.64	57.59	62.21	86.01	**88.51**	87.25	83.56	84.08	83.73

Table 9: Combined results for all the approaches for preference classification on Pixie

Approach	Model	CURRENT			NONE			OTHER			Weighted Average		
		Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
Traditional Approach	AdaBoost	71.57	73.44	72.49	45.06	35.29	39.58	63.53	68.30	71.57	63.80	64.62	64.07
	Random Forest	71.49	81.30	76.08	**64.80**	25.08	36.16	66.13	75.04	70.30	68.31	68.79	66.71
	SVM	**76.99**	**82.17**	**79.49**	62.63	**36.84**	**46.39**	**71.04**	**79.63**	**75.09**	**72.19**	**73.00**	**71.86**
Transformer Based Approach	DistilBert_{base}	82.49	87.53	84.94	61.23	52.32	56.43	80.65	80.40	80.52	77.95	78.52	78.14
	$\text{ALBERT}_{base\text{-}v2}$	87.30	87.41	87.35	58.86	57.59	58.22	82.70	83.46	83.08	80.44	80.54	80.49
	$\text{XLNET}_{base\text{-}cased}$	85.80	91.15	88.39	66.03	53.56	59.15	83.73	85.15	84.43	81.45	82.11	81.63
	RoBERTa_{base}	87.81	92.52	90.10	68.13	57.59	62.42	87.42	88.36	87.89	84.09	84.65	84.26
	RoBERTa_{large}	**88.60**	**93.02**	**90.75**	**68.99**	**61.30**	**64.92**	**88.75**	88.21	**88.48**	**85.09**	**85.49**	**85.23**
	DeBERTa_{base}	87.97	91.15	89.53	67.64	57.59	62.21	86.01	**88.51**	87.25	83.56	84.08	83.73
Prior work	ED-GAT	**83.24**	**78.05**	**80.57**	**48.89**	**54.49**	**51.54**	**76.28**	**77.79**	**77.03**	**74.44**	**73.68**	**73.99**

Results for Different Comparison Types

Table 10: Results based on type of comparisons

Sn	Model	Implicit			Explicit		
		Prec	Rec	F1	Prec	Rec	F1
1	AdaBoost	64.08	65.14	64.49	63.47	63.95	63.54
2	Random Forest	70.65	71.41	68.98	69.74	68.73	66.91
3	SVM	69.34	69.64	68.72	74.41	75.60	74.30
4	DistilBert$_{base}$	78.16	78.78	78.39	77.65	78.17	77.79
5	ALBERT$_{base\text{-}v2}$	80.12	80.58	80.32	80.62	80.49	80.54
6	XLNet$_{base}$	80.33	81.18	80.65	82.93	83.33	82.90
7	RoBERTa$_{base}$	83.51	84.21	83.74	**84.76**	85.14	84.88
8	RoBERTa$_{large}$	**84.22**	**84.76**	**84.40**	**86.17**	**86.43**	**86.26**
9	DeBERTa$_{base}$	84.09	84.76	84.28	83.01	83.20	83.02
10	ED-GAT	74.21	73.80	73.95	74.47	73.51	73.87

- We tested the consistency of our annotations and predictions by comparing with the user ratings.
- We extract user ratings for all sentences in Pixie and group them based on the preferred entity.
- The average user ratings for each group is given in the following table

Table 11: Average user ratings for different preferred entity groups

Data	Preferred Entity			
	Current	None	Other	
Entire Pixie dataset	4.656	3.321	1.993	Ground truth
Test set	4.665	3.292	1.945	Ground truth
Test set	4.608	3.139	1.991	RoBERTa predictions

Discussion

- The models struggled the most in identifying the NONE class.
  - This class was also the most ambiguous class to annotate manually
- 6.74% (120 sentences) of the test set are predicted incorrectly by all transformer-based approaches while 68.33% (1215) are predicted correctly
  - Among the wrong predictions, the majority ($\approx 62\%$) belongs to the none class, and only ($\approx 15\%$) are for implicit comparisons.
  - Among the correct predictions, the majority ($\approx 53\%$) belongs to implicit comparisons and only ($\approx 9\%$) to the none class.
- We balance the dataset and experiment with random upsampling of the minority class but did not observe any improvements in the results

Conclusion and future work

Conclusion

- We present Pixie, a new dataset containing implicit and explicit comparisons in app review and identified preferred entity
- Pixie includes comparative sentences that have been overlooked by earlier works, such as
  - Indirect comparisons
  - Implicit comparisons
- Pixie is the largest manually labeled dataset on preference classification containing ≈9k comparative sentences
- Transformer-based pretrained models fine-tuned on Pixie achieve a weighted average F1 score of 85.23% and notably outperform the previous state-of-the-art method (73.99%)
- Our preference annotations and predictions are consistent with the user ratings

Future work

- User Expectations
  - What are the features that matter most?
  - For example,
    - *If Uber had customer service that could be Lyft.*
  - Will require finer grained annotations (such as aspect of comparison)
- Subjective vs objective comparisons
  - Will aid in separating factual vs non-factual (opinion) comparisons
  - For example,
    - *I like X more than Y (subjective comparison, opinionated)*
    - *X is taller than Y (objective comparison, factual)*

"Learning to choose is hard. Learning to choose well is harder. And learning to choose well in a world of unlimited possibilities is harder still, perhaps too hard."

– Barry Schwartz, The Paradox of Choice: Why More Is Less

A. Conneau and D. Kiela.
SentEval: An evaluation toolkit for universal sentence representations.
In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova.
BERT: Pre-training of deep bidirectional transformers for language understanding.
In *Proceedings of the 17th Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

M. Ganapathibhotla and B. Liu.
Mining opinions in comparative sentences.
In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 241–248, Manchester, UK, Aug. 2008. Coling 2008 Organizing Committee.

N. Jindal and B. Liu.
Identifying comparative sentences in text documents.
In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, page 244–251, New York, NY, USA, 2006. Association for Computing Machinery.

N. Ma, S. Mazumder, H. Wang, and B. Liu.
Entity-aware dependency-based deep graph attention network for comparative preference classification.
In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5782–5788, Online, July 2020. Association for Computational Linguistics.

A. Panchenko, A. Bondarenko, M. Franzek, M. Hagen, and C. Biemann.
Categorizing comparative sentences.
In *Proceedings of the 6th Workshop on Argument Mining*, pages 136–145, Florence, Italy, Aug. 2019. Association for Computational Linguistics.

N. Reimers and I. Gurevych.
Sentence-BERT: Sentence embeddings using Siamese BERT-networks.
In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.

S. Staab and U. Hahn.
Comparatives in context.
In *Proceedings of the 14th National Conference on Artificial Intelligence*, AAAI'97/IAAI'97, page 616–621, Providence, Rhode Island, 1997. AAAI Press.

P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio.
Graph attention networks.
Sixth International Conference on Learning Representations, 2017.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le.
Xlnet: Generalized autoregressive pretraining for language understanding.
In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, Vancouver, 2019. Curran Associates, Inc.